

PIC10F220/222
数据手册

6 引脚 8 位闪存单片机

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下， Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是 “牢不可破”的。

代码保护功能处于持续发展中。 Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了 《数字器件千年版权法案 （Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip 位于美国亚利桑那州 Chandler 和 Tempe、位于俄勒冈州 Gresham 及位于加利福尼亚州 Mountain View 的全球总部、设计中心和晶圆生产厂均通过了 ISO/TS-16949:2002 认证。公司在 PICmicro® 8 位单片机、 KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外， Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

MICROCHIP

PIC10F220/222

6 引脚 8 位闪存单片机

本数据手册中包含的器件

- PIC10F220
- PIC10F222

高性能 RISC CPU

- 仅需学习 33 条单字指令
- 除程序跳转指令为双周期外，所有指令都是单周期的
- 12 位宽指令
- 2 级深硬件堆栈
- 数据和指令的直接、间接和相对寻址模式
- 8 位宽数据总线
- 8 个硬件特殊功能寄存器
- 工作速度：
  - 使用 8 MHz 内部时钟，指令周期为 500 ns
  - 使用 4 MHz 内部时钟，指令周期 1 μs

单片机特性

- 4 MHz 或 8 MHz 高精度内部振荡器：
  - 出厂时精度校准为 ±1%
- 在线串行编程 （In-Circuit Serial Programming，ICSP™）编程功能
- 支持在线调试 （In-Circuit Debugging， ICD）
- 上电复位 （Power-on Reset， POR）
- 短时器件复位定时器 （ Device Reset Timer，DRT）——典型值 1.125 ms
- 采用专用片上 RC 振荡器的看门狗定时器（Watchdog Timer ， WDT），能可靠工作
- 可编程代码保护
- 复用的 $\overline{\text{MCLR}}$ 输入引脚
- I/O 引脚上的内部弱上拉功能
- 低功耗休眠模式
- 引脚电平变化时从休眠模式唤醒

低功耗特性 /CMOS 技术

- 工作电流：
  - 当电压为 2V，频率为 4 MHz 时 < 350 μA
- 待机电流：
  - 当电压为 2V 时，典型值为 100 nA
- 低功耗、高速闪存技术：
  - 可经受 10 万次擦写
  - 数据保存时间 >40 年
- 全静态设计
- 宽工作电压范围：2.0V 到 5.5V
- 宽温度范围：
  - 工业级：-40°C 到 +85°C
  - 扩展级：-40°C 到 +125°C

外设特性

- 4 个 I/O 引脚：
  - 3 个具有独立方向控制的 I/O 引脚
  - 1 个只能输入的引脚
  - 直接驱动 LED 的高灌 / 拉电流
  - 电平变化唤醒功能
  - 弱上拉功能
- 带 8 位可编程预分频器的 8 位实时时钟 / 计数器（TMR0）
- 模数 （A/D）转换器
  - 8 位分辨率
  - 2 路外部输入通道
  - 1 路内部输入通道，专用于转换的 0.6 V 绝对参考电压

器件	程序存储器	数据存储器	I/O	8 位定时器	8 位 A/D（通道数）
	闪存 （字）	SRAM （字节）			
PIC10F220	256	16	4	1	2
PIC10F222	512	23	4	1	2

引脚图

6 引脚 SOT-23

引脚	左侧	右侧	引脚
1	GP0/AN0/ICSPDAT	GP3/MCLR/VPP	6
2	VSS	VDD	5
3	GP1/AN1/ICSPCLK	GP2/T0CKI/FOSC4	4

PIC10F220/222

8 引脚 PDIP

引脚	左侧	右侧	引脚
1	N/C	GP3/MCLR/VPP	8
2	VDD	VSS	7
3	GP2/T0CKI/FOSC4	N/C	6
4	GP1/AN1/ICSPCLK	GP0/AN0/ICSPDAT	5

PIC10F220/222

初稿

目录

致 客 户

我们旨在提供最佳文档供客户正确使用 Microchip 产品。为此，我们将不断改进出版物的内容和质量，使之更好地满足您的要求。出版物的质量将随新文档及更新版本的推出而得到提升。

如果您对本出版物有任何问题和建议，请通过电子邮件联系我公司 TRC 经理，电子邮件地址为 **CTRC@microchip.com**，或将本数据手册后附的 **《读者反馈表》**传真到 86-21-5407 5066。我们期待您的反馈。

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

http://www.microchip.com

查看数据手册中任意一页下边角处的文献编号即可确定其版本。文献编号中数字串后的字母是版本号，例如：DS30000A是DS30000 的 A 版本。

勘误表

现有器件可能带有一份勘误表，描述了实际运行与数据手册中记载内容之间存在的细微差异以及建议的变通方法。一旦我们了解到器件 / 文档存在某些差异时，就会发布勘误表。勘误表上将注明其所适用的硅片版本和文件版本。

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

- Microchip 网站 http://www.microchip.com
- 当地 Microchip 销售办事处 （见最后一页）

在联络销售办事处时，请说明您所使用的器件型号、硅片版本和数据手册版本 （包括文献编号）。

客户通知系统

欲及时获知 Microchip 产品的最新信息，请到我公司网站 **www.microchip.com** 上注册。

注：

1.0 一般说明

Microchip Technology 的 PIC10F220/222 器件是一款低成本、高性能的 8 位全静态闪存 CMOS 单片机。它采用 RISC 架构，仅有 33 条单字单周期指令。除需要两个周期的程序跳转指令外，所有指令都是单周期指令（4 MHz 时为 1 μs）。PIC10F220/222 器件的性能大大高于同等价位的其他产品。12 位宽的指令具有高度的对称性，与同类的 8 位单片机相比其代码压缩了两倍。易于使用和记忆的指令集，大大缩短了开发周期。

PIC10F220/222产品装备的特殊功能降低了系统成本和功耗。上电复位 （POR）和器件复位定时器 （DRT）使其不再需要外部复位电路。在该器件上提供了内部振荡器 （INTOSC）模式，从而节省了有限的 I/O 引脚，使其可作其他用途。节省功耗的休眠模式、看门狗定时器和代码保护功能降低了系统的成本和功耗，同时提高了系统的可靠性。

PIC10F220/222 器件具有低成本的闪存存储器，适用于任意数量的批量生产。用户可以完全利用 Microchip 在闪存可编程单片机市场的价格领导地位，同时得益于闪存可编程产品的灵活性。

支持 PIC10F220/222 器件的开发工具有：全功能宏汇编器、软件模拟器、在线调试器、C 编译器、低成本开发编程器和全功能编程器。IBM® PC 和兼容机都能支持这些工具。

1.1 应用

PIC10F220/222器件适合从个人护理设备和安防系统到低功耗远程收发器。闪存技术可以非常快速和方便地定制应用程序（发送器代码、设备设置、接收器频率等）。小型封装的器件可用于通孔或表面封装，使这些单片机可以完全适应有空间限制的应用。低成本、低功耗、高性能、使用简便和 I/O 灵活性使 PIC10F220/222 器件能够在您认为根本不能使用单片机的场合也可以大显神通（如较大系统中的定时器功能、逻辑电路和 PLD 以及协处理器应用）。

表 1-1： PIC10F220/222 器件 (1, 2)

		PIC10F220	PIC10F222
时钟	最高工作频率 （MHz）	8	8
存储器	闪存程序存储器	256	512
	数据存储器 （字节）	16	23
外设	定时器模块	TMR0	TMR0
	引脚电平变化时从休眠模式唤醒	有	有
	模拟输入	2	2
特性	I/O 引脚	3	3
	只能输入的引脚	1	1
	内部上拉	有	有
	在线串行编程	有	有
	指令数	33	33
	封装	6 引脚 SOT-23， 8 引脚 PDIP	6 引脚 SOT-23， 8 引脚 PDIP

注 1： PIC10F220/222 器件具有上电复位、可选看门狗定时器、可选代码保护、高 I/O 电流驱动能力和高精度内部振荡器。
2： PIC10F220/222 使用数据引脚 GP0 和时钟引脚 GP1 实现串行编程。

注：

2.0 器件种类

器件提供不同的封装类型。根据应用和产品要求，可以使用本章中的信息选择正确的器件。在下订单时，请使用本数据手册后面的PIC10F220/222产品标识体系来指定正确的器件编号。

2.1 快速批量编程 （QTP）器件

Microchip 为工厂生产订单提供快速批量编程 （Quick Turn Programming，QTP）服务。此服务适用于那些不想为中到大批量器件进行编程并且自有的代码已稳定的用户。此器件与闪存器件相同，只是所有闪存单元和熔丝选项已在出厂前预先编程了。批量产品交付前经过了某些代码和原型校验。欲知更多详情，请与当地的 Microchip 销售办事处联系。

2.2 带序列号的快速批量编程 （SQTP^SM）器件

Microchip 提供这种独特的编程服务，可通过编程为每个器件中一些用户定义的单元指定不同的序列号。这些序列号可以是随机的、伪随机或连续的。

串行编程使每个器件都具有一个惟一的编号，可作为登录码、密码或 ID 编号。

注：

3.0 架构概述

PIC10F220/222 系列的高性能可以归功于 RISC 单片机中普遍采用的许多架构特点。首先，PIC10F220/222 采用了哈佛架构，在这种架构中，分别使用独立的总线访问程序和数据存储器。与传统的程序和数据存储共用同一条总线的冯 • 诺依曼架构相比，哈佛架构具有更加优良的总线带宽。独立的程序和数据存储器使得指令宽度不再是 8 位。 由于指令操作码为 12 位宽，所以所有指令都可以是单字指令。通过 12 位宽的程序存储器总线可以在单周期内取一条 12 位的指令。两级流水线可以使取指和执指同时进行，因此，除了程序跳转指令外，所有指令（33 条）都可以在单周期（4 MHz 时为 1 µs，或 8 MHz 时为 500 ns）内执行。

下表列出了 PIC10F220/222 器件的程序存储器（闪存）和数据存储器 （RAM）。

器件	存储器	
	程序	数据
PIC10F220	256 x 12	16 x 8
PIC10F222	512 x 12	23 x 8

PIC10F220/222能直接或间接寻址其文件寄存器或数据存储器。所有特殊功能寄存器（Special Function Register，SFR），包括程序计数器，都映射到数据存储空间。PIC10F220/222 有一个高度正交 （对称）的指令集，因此它可以使用任何寻址模式对任一寄存器执行任何操作。指令集的对称特性以及无 “特别理想状态” 让使用 PIC10F220/222 编程更简单有效。此外，学习时间也得以显著缩短。

PIC10F220/222 器件包含一个 8 位 ALU 和工作寄存器。ALU 是一个通用算术单元。它对工作寄存器和文件寄存器中的数据进行算术和布尔运算。

ALU 为 8 位宽，能够进行加、减、移位和逻辑操作。除非特别指明，否则算术运算一般是以 2 的补码的形式进行的。在两个操作数的指令中，典型情况下，其中的一个操作数是在 W（工作）寄存器中，另一个操作数为文件寄存器或是一个立即数。在单操作数指令中，操作数为 W 寄存器或某个文件寄存器。

W 寄存器是一个 8 位宽、用于 ALU 运算的工作寄存器。该寄存器不可寻址。

根据所执行的指令，ALU 可以影响状态寄存器中的进位标志位 C、半进位标志位 DC 和全零标志位 Z。在减法运算中，C 和 DC 位分别作为借位和半借位标志位。例如指令 SUBWF 和 ADDWF。

图 3-1 给出了简化框图，表 3-1 给出了相应器件的引脚配置。

图 3-1： 框图

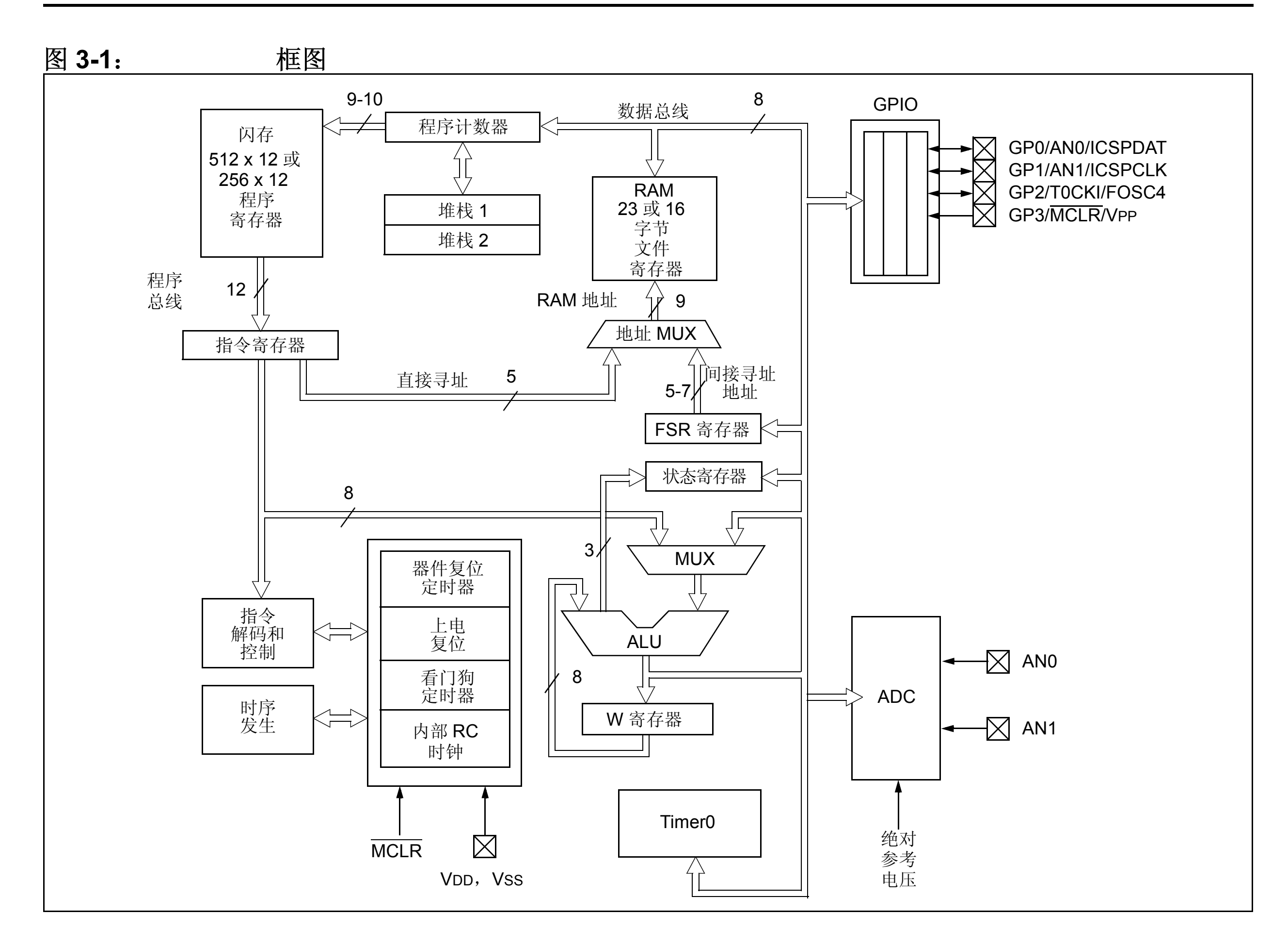


表 3-1： 引脚配置说明

名称	功能	输入类型	输出类型	说明
GP0/AN0/ICSPDAT	GP0	TTL	CMOS	双向 I/O 引脚。可用软件编程为内部弱上拉和当引脚电平变化时从休眠唤醒。
	AN0	AN	—	模拟输入。
	ICSPDAT	ST	CMOS	在线编程数据。
GP1/AN1/ICSPCLK	GP1	TTL	CMOS	双向 I/O 引脚。可用软件编程为内部弱上拉和当引脚电平变化时从休眠唤醒。
	AN1	AN	—	模拟输入。
	ICSPCLK	ST	CMOS	在线编程时钟。
GP2/T0CKI/FOSC4	GP2	TTL	CMOS	双向 I/O 引脚。
	T0CKI	ST	—	时钟输入到 TMR0。
	FOSC4	—	CMOS	振荡器 /4 输出。
GP3/MCLR/VPP	GP3	TTL	—	输入引脚。可用软件编程为内部弱上拉和当引脚电平变化时从休眠唤醒。
	MCLR	ST	—	主复位。当配置为 MCLR 时，该引脚为低电平时复位器件 （低电平有效）。在器件正常工作期间 MCLR/VPP 上的电压不能超过 VDD，否则器件会进入编程模式。如果配置为MCLR，则始终使能弱上拉功能。
	VPP	HV	—	编程电压输入。
VDD	VDD	P	—	逻辑电路和 I/O 引脚的正电源。
VSS	VSS	P	—	逻辑电路和 I/O 引脚的参考地。

图注： I = 输入， O = 输出， I/O = 输入 / 输出， P = 电源， — = 未用， TTL = TTL 输入，
ST = 施密特触发器输入， AN = 模拟输入

3.1 时钟机制 / 指令周期

时钟信号在器件内部被四分频后，产生四个不重叠的正交时钟节拍，称为 Q1、Q2、Q3 和 Q4。在每个 Q1 节拍 PC 递增；在 Q4 节拍从程序存储器取指并将指令锁存到指令寄存器。指令的译码和执行是在下一个 Q1 到 Q4 节拍中完成的。图 3-2 和例 3-1 给出了时钟和指令执行的流程图。

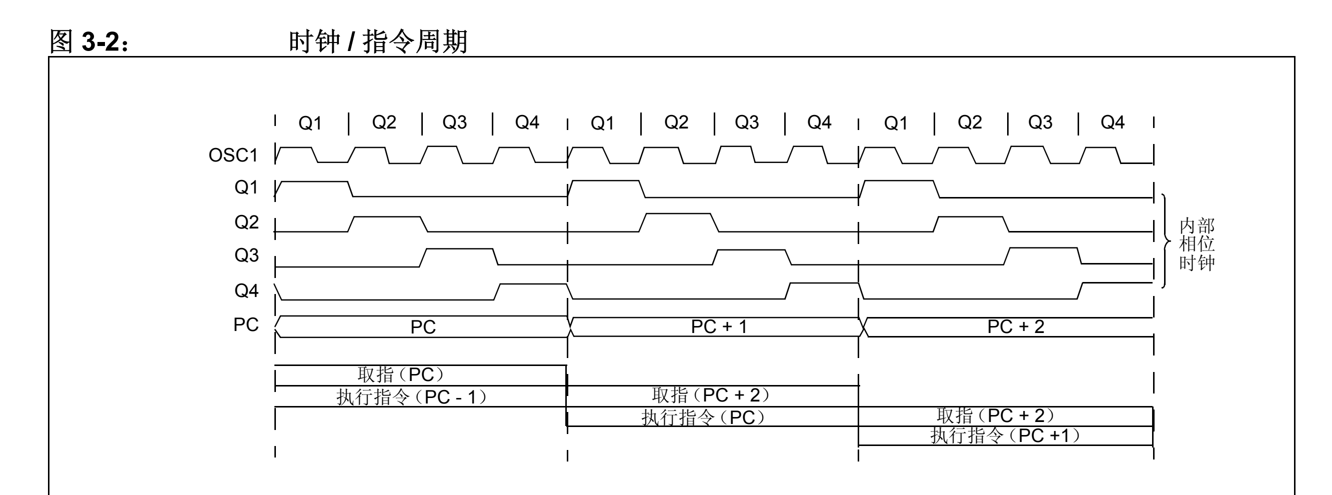
3.2 指令流 / 流水线

一个指令周期由四个 Q 节拍组成（Q1、Q2、Q3 和 Q4）。取指和执行指令是流水线操作的，因此取指需要一个指令周期，而译码和执行指令则需要另一个指令周期。但由于是流水线操作，所以每条指令的等效执行时间都是一个指令周期。如果某条指令改变了程序计数器（如 GOTO 指令），则需要两个指令周期才能完成该指令（见例 3-1）。

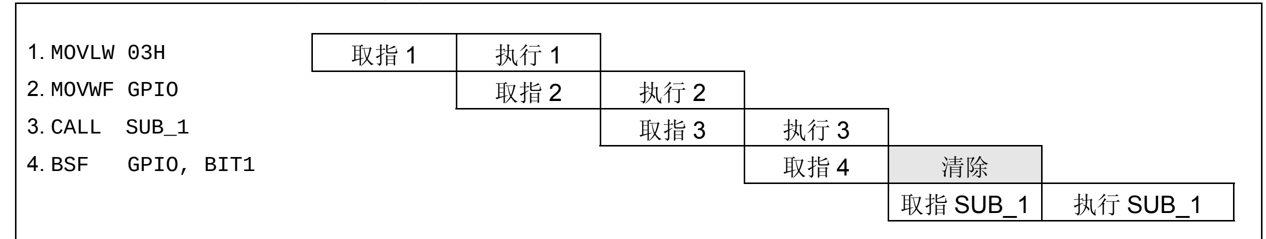
取指周期由 Q1 节拍中 PC 加 1 开始。

指令的执行过程如下：在 Q1 节拍，将所取指令锁存到指令寄存器中。然后在 Q2、Q3 和 Q4 节拍中进行指令的译码和执行。其中读数据存储器（读操作数）发生在 Q2 节拍，写操作发生在 Q4 节拍（写目标单元）。

图 **3-2**： 时钟 / 指令周期



例 **3-1**： 指令流水线流程图



除程序跳转指令之外，所有指令都是单周期指令。由于程序跳转指令将导致一条已取指令从流水线清除，需要重新取指，然后执行指令，所以程序跳转指令需要两个周期。

注：

4.0 存储器构成

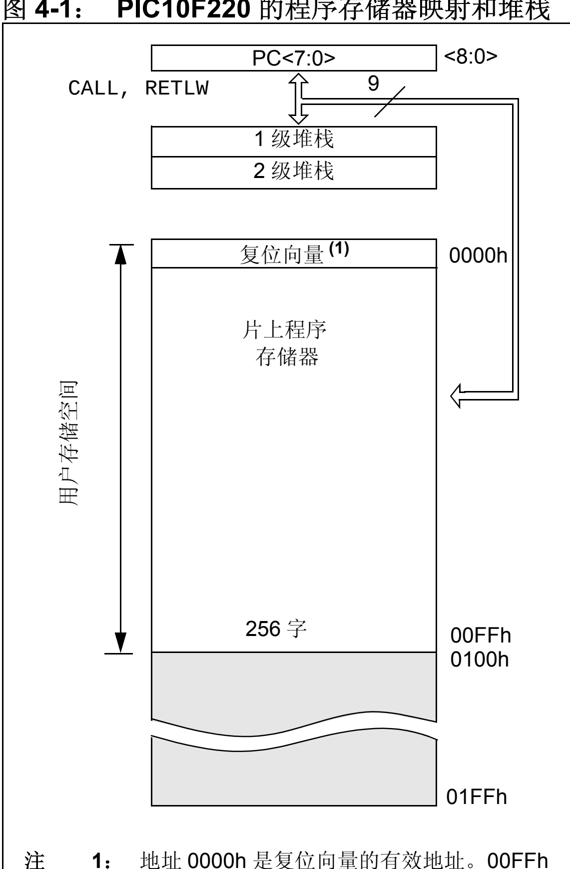
PIC10F220/222 的存储器分为程序存储器和数据存储器。可使用文件选择寄存器（File Selection Register，FSR）来访问数据存储器分区。

4.1 PIC10F220 的程序存储器构成

PIC10F220 器件具有 9 位程序计数器（Program Counter，PC），可以对512 x 12程序存储空间进行寻址。

PIC10F220 中只有前 256 x 12（0000h-00FFh）空间是物理实现的（见图 4-1）。访问这些地址边界以外的单元将折回到前 256 x 12 空间（PIC10F220）。有效的复位向量位于 0000h，（见图 4-1）。00FFh 单元（PIC10F220）包含内部时钟振荡器的校准值。决不能改写该值。

图 4-1： PIC10F220 的程序存储器映射和堆栈



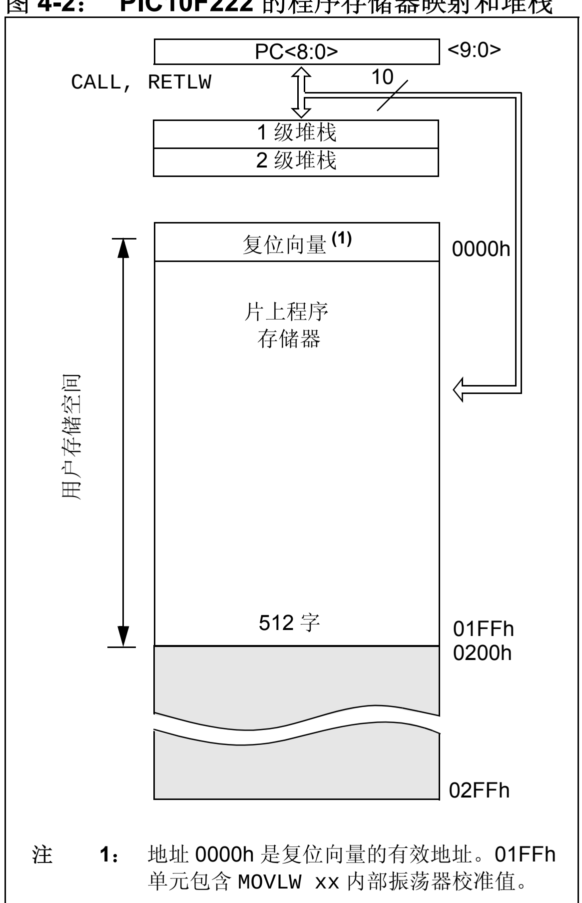
注 1： 地址 0000h 是复位向量的有效地址。00FFh 单元包含 MOVLW xx 内部振荡器校准值。

4.2 PIC10F222 的程序存储器构成

PIC10F222 器件具有 10 位程序计数器（PC），可以对 1024 x 12 程序存储空间进行寻址。

该器件程序存储器的高地址部分中，只有前 512 x 12（0000h-01FFh）空间是物理实现的（见图 4-2）。访问这些地址边界之上的单元将折回到前 256 x 12 空间（PIC10F222）。有效的复位向量位于0000h（见图4-2）。01FFh 单元（PIC10F222）包含内部时钟振荡器的校准值。决不能改写该值。

图 4-2： PIC10F222 的程序存储器映射和堆栈



注 1： 地址 0000h 是复位向量的有效地址。01FFh 单元包含 MOVLW xx 内部振荡器校准值。

4.3 数据存储器构成

数据存储器由寄存器（RAM 字节）组成。因此，器件的数据存储器由它的文件寄存器指定。文件寄存器被分为两个功能组：特殊功能寄存器（SFR）和通用寄存器（General Purpose Registers， GPR）。

特殊功能寄存器包括 TMR0 寄存器、程序计数器（PC）、状态寄存器、I/O 寄存器（GPIO）和文件选择寄存器（File Select Register， FSR）。此外，特殊功能寄存器也用于控制 I/O 端口配置和预分频比选择。

通用寄存器用于存储指令执行所需的数据和控制信息。

对于 PIC10F220 来说，文件寄存器由 9 个特殊功能寄存器和 16 个通用寄存器组成（图 4-3 和图 4-4）。

对于 PIC10F222 来说，文件寄存器由 9 个特殊功能寄存器和 23 个通用寄存器组成（图 4-4）。

4.3.1 通用文件寄存器

可以直接访问通用文件寄存器，也可以通过文件选择寄存器（FSR）间接访问该寄存器。请参见**第 4.9 节“间接数据寻址：INDF 和 FSR 寄存器”**。

图 4-3： PIC10F220 文件寄存器映射

文件地址	
00h	INDF(1)
01h	TMR0
02h	PCL
03h	STATUS
04h	FSR
05h	OSCCAL
06h	GPIO
07h	ADCON0
08h	ADRES
09h–0Fh	未用 (2)
10h–1Fh	通用寄存器

注 1： 这不是物理寄存器。请参见**第 4.9 节“间接数据寻址：INDF 和 FSR 寄存器”**。

2： 未用，读为 00h。

图 4-4： PIC10F222 文件寄存器映射



注 1： 这不是物理寄存器。请参见**第 4.9 节“间接数据寻址：INDF 和 FSR 寄存器”**。

 初稿

4.3.2 特殊功能寄存器

特殊功能寄存器 （SFR）是 CPU 和外设功能用来控制器件操作的寄存器 （表 4-1）。

特殊功能寄存器可以被分成两组。本章仅讲述与 “内核”有关的特殊功能寄存器。那些与外设功能部件的操作有关的特殊功能寄存器将在相应的外设功能部件章节中讲述。

表 4-1： 特殊功能寄存器 （SFR）汇总

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值[2]	页码
00h	INDF	使用 FSR 中的内容对数据存储器进行寻址 （不是物理寄存器）								xxxx xxxx	20
01h	TMR0	8 位实时时钟 / 计数器								xxxx xxxx	25
02h	PCL[1]	PC 的低 8 位								1111 1111	19
03h	STATUS	GPWUF	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0--1 1xxx[3]	16
04h	FSR	间接数据存储器地址指针								111x xxxx	20
05h	OSCCAL	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	FOSC4	1111 1110	18
06h	GPIO	—	—	—	—	GP3	GP2	GP1	GP0	---- xxxx	21
07h	ADCON0	ANS1	ANS0	—	—	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	11-- 1100	30
08h	ADRES	模数转换的结果								xxxx xxxx	30
N/A	TRISGPIO	—	—	—	—	I/O 控制寄存器				---- 1111	23
N/A	OPTION	$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	17

图注： - = 未用，读为 0， x = 未知， u = 不变。

注 1： 无法直接访问程序计数器的高字节。请参见**第 4.7 节 “程序计数器”**了解如何访问这些位。

2： 其他复位 （除上电复位外）包括通过 $\overline{\text{MCLR}}$ 引脚的外部复位、看门狗定时器复位和引脚电平变化时唤醒复位。

3： 请参见表 8-1 了解其他复位的特定值。

4.4 状态寄存器

此寄存器包含ALU的算术状态、复位状态和页面预选择位。

状态（STATUS）寄存器和任何其他寄存器一样，可以作为任何指令的目标寄存器。如果一条影响 Z、DC 或 C 位的指令的目标寄存器是状态寄存器，则会禁止对这 3 位进行写操作。根据器件逻辑，这些位会被置 1 或清零。而且，TO 和 PD 位也是不可写的。所以当执行一条把状态寄存器作为目标寄存器的指令后，状态寄存器的结果可能和预想的不一样。

例如，`CLRF Status` 将把高三位清零并将 Z 位置 1。这将把状态寄存器变为 `000u u1uu`（其中 u = 不变）。

因此，建议仅使用 `BCF`、`BSF` 和 `MOVWF` 指令改变来改变状态寄存器，因为这些指令不会影响状态寄存器中的 Z、DC 或 C 位。欲知其他影响状态位的指令，请参见**第 9.0 节 “指令集综述”**。

寄存器 4-1： 状态寄存器（地址：**03h**）

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
GPWUF	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
bit 7							bit 0

bit 7 **GPWUF**：GPIO 复位位
- 1 = 由于引脚电平发生变化从休眠唤醒而复位
- 0 = 上电复位或其他复位

bit 6-5 **保留**：不使用。使用该位会影响与今后产品的向上兼容性。

bit 4 **$\overline{\text{TO}}$**：超时位
- 1 = 上电后、执行 `CLRWDT` 指令或 `SLEEP` 指令后
- 0 = 发生了 WDT 超时

bit 3 **$\overline{\text{PD}}$**：掉电位
- 1 = 上电后或执行 `CLRWDT` 指令后
- 0 = 执行 `SLEEP` 指令后

bit 2 **Z**：零标志位
- 1 = 算术运算或逻辑运算结果为零
- 0 = 算术运算或逻辑运算结果不为零

bit 1 **DC**：半进位 / $\overline{\text{借位}}$标志位（对于 `ADDWF` 和 `SUBWF` 指令）

ADDWF：
- 1 = 结果的第 4 个低位向高位发生了进位
- 0 = 结果的第 4 个低位没有向高位发生进位

SUBWF：
- 1 = 结果的第 4 个低位没有发生借位
- 0 = 结果的第 4 个低位发生了借位

bit 0 **C**：进位 / $\overline{\text{借位}}$标志位（对于 `ADDWF`、`SUBWF`、`RRF` 和 `RLF` 指令）

ADDWF：	SUBWF：	RRF 或 RLF：
1 = 发生了进位	1= 没有发生借位	分别装载最低位或最高位
0 = 没有发生进位	0 = 发生了借位	

图注：			
R = 可读位	W = 可写位	U= 未用位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

4.5 选项寄存器

选项（OPTION）寄存器是 8 位宽的只写寄存器，它包含各种用于配置 Timer0/WDT 预分频器和 Timer0 的控制位。

选项寄存器不是存储器映射的，通过执行 `OPTION` 指令可将 W 寄存器的内容传送到选项寄存器。复位将使 OPTION<7:0> 位置 1。

> 注 1：如果 TRIS 位清零（= `0`），将禁止相应引脚的电平变化时唤醒功能和上拉功能（即，注意 TRIS 将改写选项寄存器中的 GPPU 和 GPWU 控制位）。
>
> 2：如果将 T0CS 位置 1，它将改写 T0CKI 引脚的 TRIS 功能。

寄存器 4-2： **选项寄存器（PIC10F220/222）**

W-1	W-1	W-1	W-1	W-1	W-1	W-1	W-1
$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

bit 7 **$\overline{\text{GPWU}}$**：引脚电平变化唤醒使能位（GP0、 GP1 和 GP3）
`1` = 禁止
`0` = 使能

bit 6 **$\overline{\text{GPPU}}$**：弱上拉使能位（GP0、 GP1 和 GP3）
`1` = 禁止
`0` = 使能

bit 5 **T0CS**：Timer0 时钟源选择位
`1` = T0CKI 引脚上的传输信号（优先级高于 T0CKI 引脚上的 TRIS）
`0` = 内部指令周期时钟 Fosc/4

bit 4 **T0SE**：Timer0 时钟源边沿选择位
`1` = 在 T0CKI 引脚上电平的下降沿递增
`0` = 在 T0CKI 引脚上电平的上升沿递增

bit 3 **PSA**：预分频器分配位
`1` = 预分频器分配给 WDT
`0` = 预分频器分配给 Timer0

bit 2-0 **PS<2:0>**：预分频器分频比选择位

位值	Timer0 分频比	WDT 分频比
`000`	1 : 2	1 : 1
`001`	1 : 4	1 : 2
`010`	1 : 8	1 : 4
`011`	1 : 16	1 : 8
`100`	1 : 32	1 : 16
`101`	1 : 64	1 : 32
`110`	1 : 128	1 : 64
`111`	1 : 256	1 : 128

图注：			
R = 可读位	W = 可写位	U= 未用位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

4.6 振荡器校准寄存器

振荡器校准 （OSCCAL）寄存器用于校准内部高精度 4 MHz/8 MHz 振荡器。它包含 7 位校准位。

注：	擦除器件同时也会擦除预编程的内部振荡器内部校准值。必须在擦除该部分之前读取校准值，这样以后才可以正确地重新编程。

在您读取了校准常数以后，请不要更改该值。请参见**第 8.2.2 节 “内部 4 MHz/8 MHz 振荡器”**。

寄存器 4-3：　OSCCAL：振荡器校准寄存器 （地址：05h）

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0
CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	FOSC4
bit 7							bit 0

bit 7-1　**CAL<6:0>**：振荡器校准位

0111111 = 最高频率
•
•
•
0000001
0000000 = 中心频率
1111111
•
•
•
1000000 = 最低频率

bit 0　**FOSC4**：INTOSC/4 输出使能位 [(1)]

1 = INTOSC/4 输出到 GP2
0 = GP2/T0CKI/COUT 用作 GP2 功能

注　**1**：如果使能将改写控制 GP2/T0CKI/COUT 引脚的寄存器。

图注：			
R = 可读位	W = 可写位	U = 未用位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

4.7 程序计数器

当执行程序指令时，程序计数器 （PC）中将包含下一条要被执行的程序指令的地址。除更改 PC 的指令外，每执行一条指令， PC 值都会增 1。

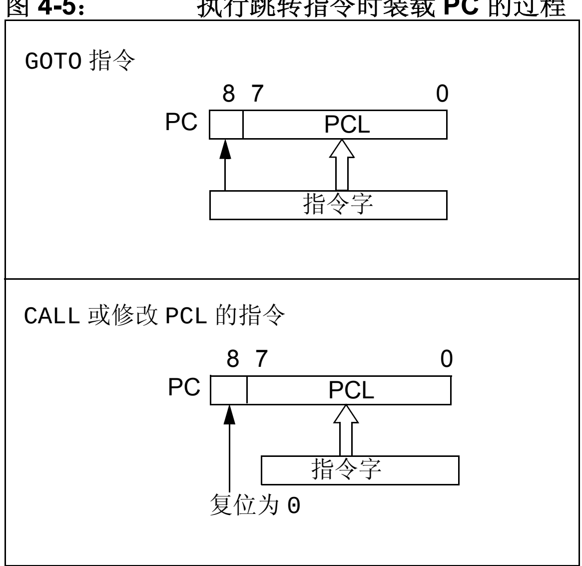
对于 GOTO 指令，PC 的 bit 0 到 bit 8 由 GOTO 指令字提供。PC 锁存器 （PCL）被映射到 PC<7:0>。

对于 CALL 指令或其他以 PCL 作为目标寄存器的指令，PC 的 bit 0 到 bit 7 也是由指令字提供的。但是，PC<8> 不由指令字提供，它始终保持清零 （图 4-5）。

以 PCL 为目标寄存器的指令或修改 PCL 的指令包括 MOVWF PC、ADDWF PC 和 BSF PC,5。

> **注：** 由于 CALL 指令或任何修改 PCL 的指令会将 PC<8> 清零，因此所有子程序调用或相对跳转指令都被限制在任何程序存储器页（512 字长）的前 256 个单元中。

图 4-5： 执行跳转指令时装载 PC 的过程



4.7.1 复位的影响

复位时 PC 被置为全 1，这意味着 PC 指向程序存储器中的最后一个单元（即振荡器校准指令）。执行 MOVLW XX 指令后，PC 将计满返回到 0000h 单元并开始执行用户代码。

4.8 堆栈

PIC10F220 器件有一个 2 级深、 8 位宽的硬件 PUSH/POP 堆栈。

PIC10F222 器件有一个 2 级深、 9 位宽的硬件 PUSH/POP 堆栈。

执行 CALL 指令会将堆栈第一级单元中的内容压入（PUSH）第二级单元中，然后将 PC 的当前值加 1 压入堆栈的第一级。如果连续执行了多于两条 CALL 指令，只会存储最近的两个返回地址。

执行 RETLW 指令会将堆栈第一级单元中的内容弹出（POP）到 PC，然后将堆栈第 2 级单元中的内容复制到第一级单元中。如果连续执行了多于两条 RETLW 指令，将使用先前存储在堆栈第二级单元中的地址填充堆栈。

> **注 1：** 可以使用在指令中指定的立即数装载W寄存器。这对于在程序存储器中实现数据查找表尤其有用。
>
> **2：** 不存在指示堆栈上溢或下溢条件的状态位。
>
> **3：** 不存在称为 PUSH 或 POP 的指令助记符。这些是执行 CALL 和 RETLW 指令后出现的情形。

4.9 间接数据寻址：INDF 和 FSR 寄存器

INDF 寄存器不是物理寄存器。寻址 INDF 实际寻址的是地址包含在 FSR 寄存器中的寄存器（FSR 是一个*指针*）。这就是间接寻址。

4.9.1 间接寻址

- 地址为 09 的文件寄存器包含值 10h
- 地址为 0A 的文件寄存器包含值 0Ah
- 将值 09 载入 FSR 寄存器
- 读 INDF 寄存器将返回值 10h
- 将 FSR 寄存器的值增 1 （FSR = 0A）
- 读 INDF 寄存器将返回值 0Ah。

直接读 INDF 本身 （FSR = `0`）将返回 00h。直接写 INDF 寄存器将导致空操作 （虽然会影响状态位）。

例 4-1 给出了使用间接寻址将 RAM 单元 10h-1Fh 清零的简单程序。

例 4-1：　使用间接寻址清零 RAM 的方法

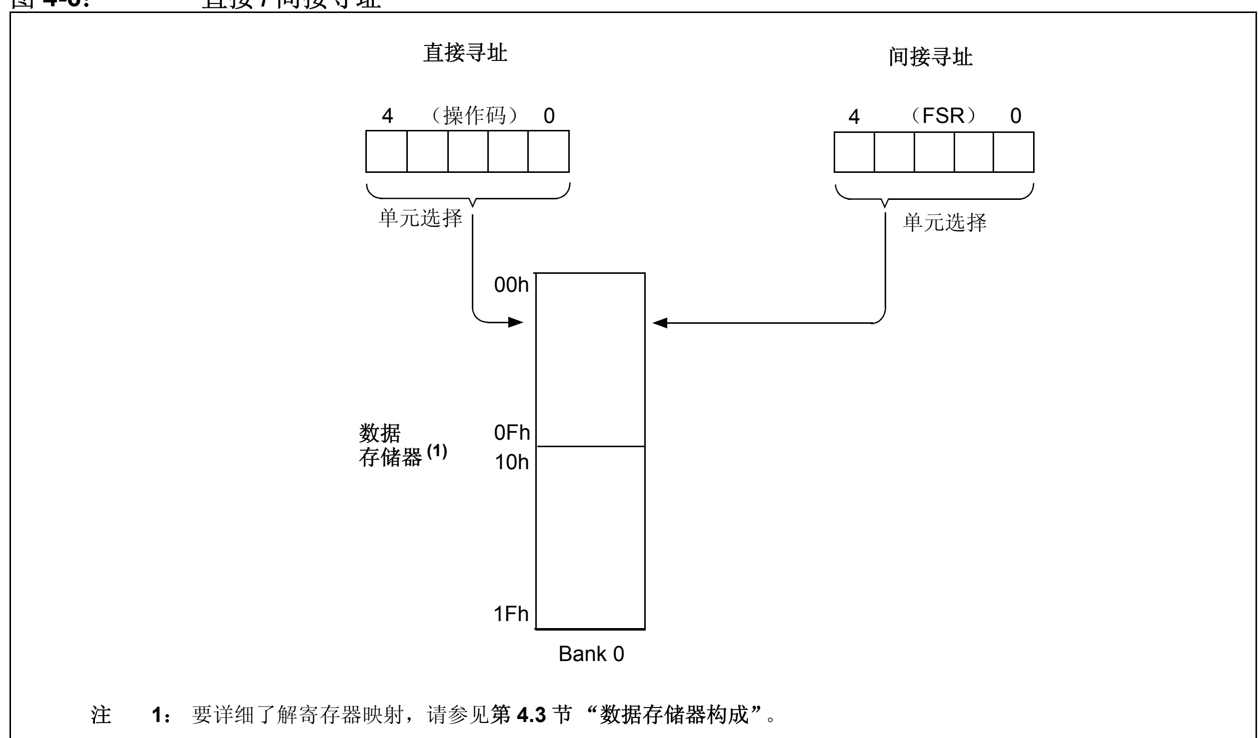
```
        MOVLW   0x10    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF
                        ;register
        INCF    FSR, F  ;inc pointer
        BTFSC   FSR, 4  ;all done?
        GOTO    NEXT    ;NO, clear next
CONTINUE
            :           ;YES, continue
            :
```

FSR 是一个 5 位宽的寄存器。它与 INDF 寄存器配合使用来间接寻址数据存储区。

FSR<4:0> 位被用来选择数据存储器地址 00h 到 1Fh。

注： 10F220和10F222不使用分区。FSR<7:5> 不被使用并且读为 `1`。

图 4-6：　直接 / 间接寻址



注 1：要详细了解寄存器映射，请参见**第 4.3 节 “数据存储器构成”**。

5.0 I/O 端口

和任何其他寄存器一样， I/O 寄存器也可在程序控制下进行读写。但是，读指令（如 `MOVF PORTB,W`）始终读取I/O引脚，而与引脚是处于输入还是输出模式无关。复位时，由于 I/O 控制寄存器被置为全 1，因此所有 I/O 端口都被定义为输入（输入为高阻态）。

5.1 GPIO

GPIO 是一个 8 位 I/O 寄存器。仅使用其低 4 位（GP<3:0>）。bit 4 到 bit 7 位未用，读为 0。注意 GP3 是一个只能输入的引脚。GP0、 GP1 和 GP3 引脚可以被配置为具有内部弱上拉和电平变化唤醒功能。电平变化唤醒和弱上拉功能并不可通过引脚选择。如果 GP3/MCLR 被配置为 MCLR，可通过配置字使能弱上拉，但禁止该引脚的电平变化唤醒功能。

5.2 TRIS 寄存器

通过执行 `TRIS f` 指令，可使用 W 寄存器的内容装载输出驱动器控制寄存器。将 TRIS 寄存器的某位置 1，可使相应的输出驱动器置于高阻模式。而将 TRIS 寄存器的某位清零，会将输出数据锁存器的内容送入选定的引脚并使能输出缓冲器。特殊的引脚为 GP3 引脚和 GP2/T0CKI/FOSC4引脚，前者为只输入引脚，后者是由各种寄存器控制的引脚。请参见表 5-1。

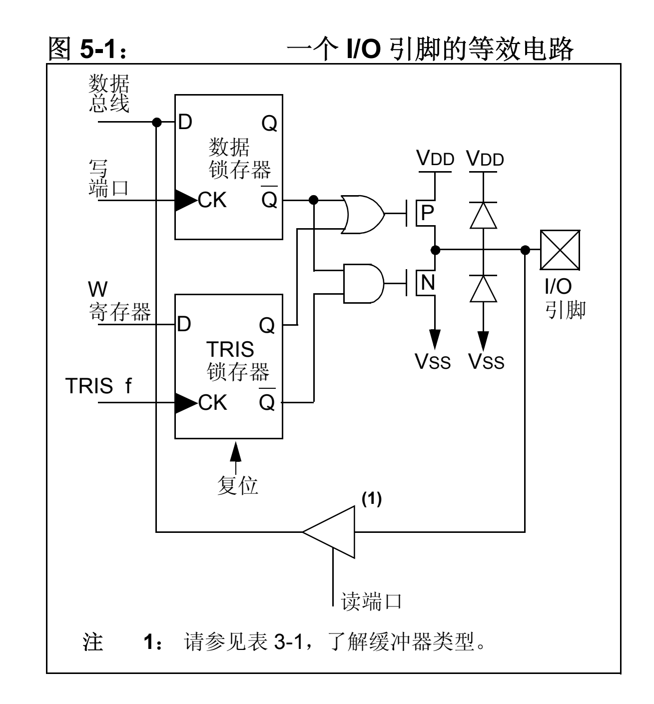
注：	读端口读的是引脚而非输出数据锁存器。也就是说，如果使能了某个引脚的输出驱动器，并驱动为高电平，但外部系统将该引脚保持为低电平，那么读相应端口将表明该引脚为低电平。

TRIS 寄存器是 “只写”的，并且在复位时被置为全 1（禁止输出驱动器）。

5.3 I/O 接口

图 5-5 给出了 I/O 端口引脚的等效电路。除只能输入的 GP3 引脚外，所有端口引脚都可用于输入和输出操作。对于输入操作，这些端口是不锁存的。所有输入数据都必须位于引脚上，直到被输入指令（如 `MOVF GPIO, W`）读取为止。输出是锁存的并将保持不变，直到输出锁存器被再次写入为止。要将端口引脚用作输出，必须将 TRIS 寄存器中相应的方向控制位清零（= 0）。要将端口引脚用作输入，必须将相应的 TRIS 位置 1。可分别将任何 I/O 引脚（除 GP3）编程为输入或输出。

图 5-1： 一个 I/O 引脚的等效电路



注 1：请参见表 3-1，了解缓冲器类型。

表 5-1： 引脚功能的优先级

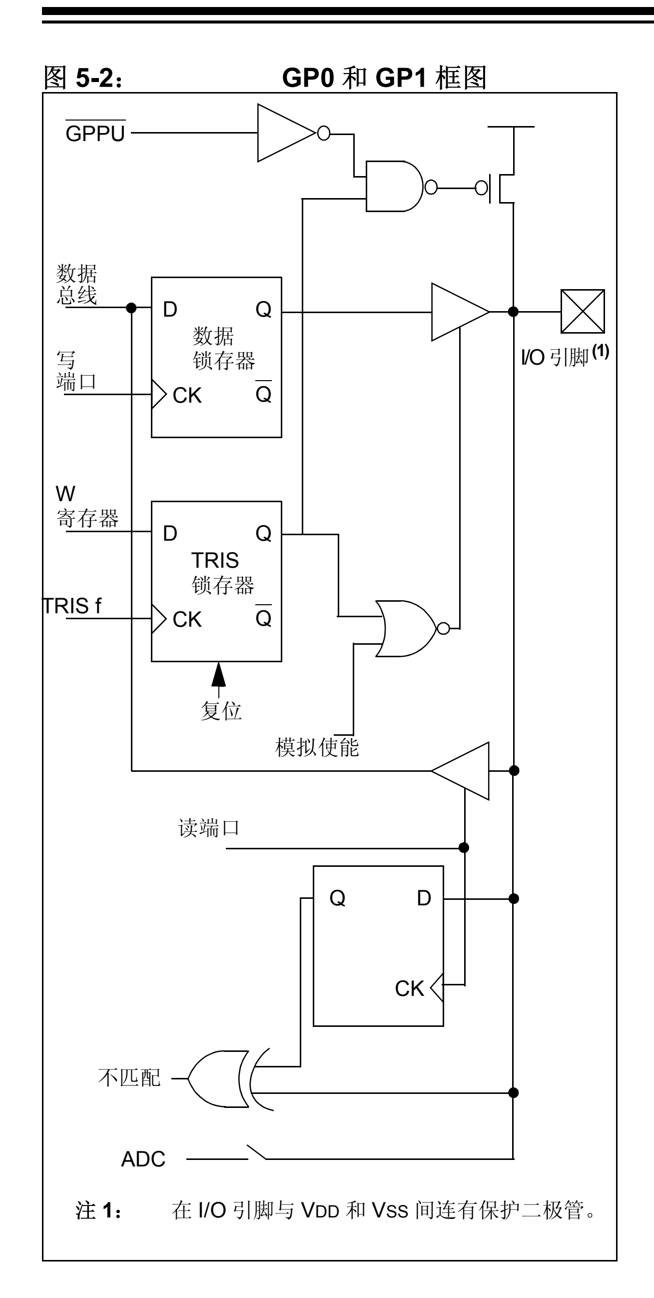
优先级	GP0	GP1	GP2	GP3
1	AN0	AN1	FOSC4	I/$\overline{MCLR}$
2	TRIS GPIO	TRIS GPIO	T0CKI	—
3	—	—	TRIS GPIO	—

表 5-2： 引脚配置为数字模式要求

	FOSC4	T0CS	ANS1	ANS0	MCLRE
寄存器	OSCCAL	OPTION	ADCON0	ADCON0	CONFIG
GP0	—	—	—	0	—
GP1	—	—	0	—	—
GP2	0	0	—	—	—
GP3	—	—	—	—	0

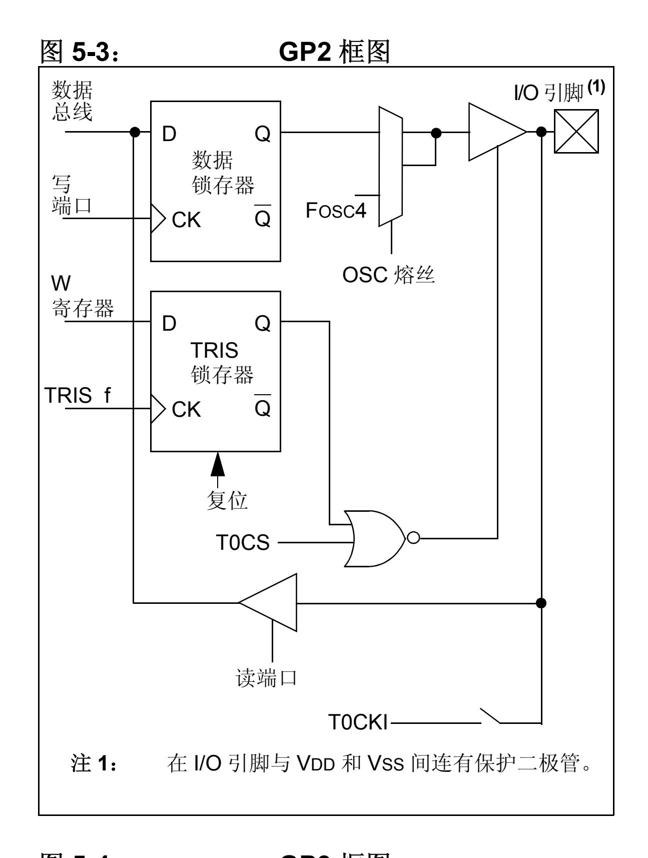
图注： — = 位状态对引脚配置为数字模式没有影响。

图 5-2： GP0 和 GP1 框图



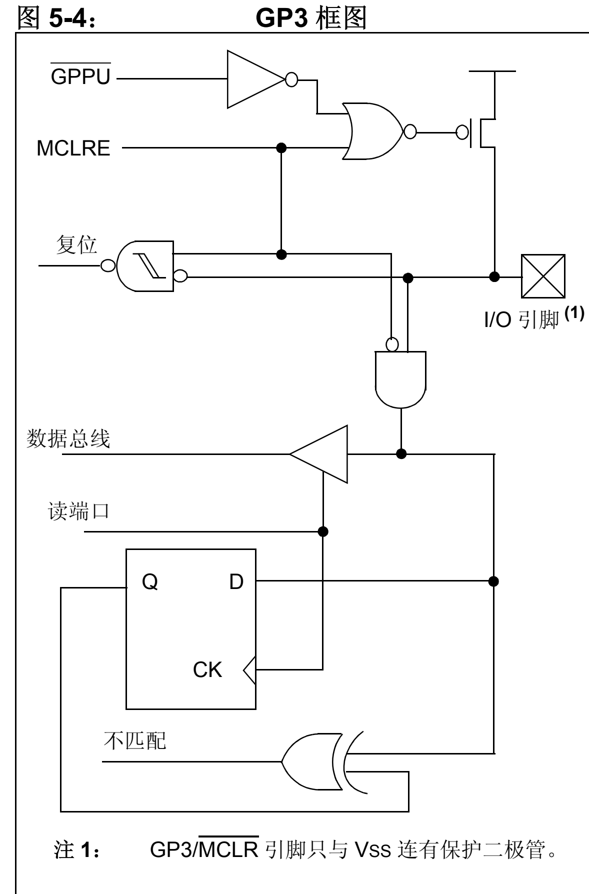
注 1： 在 I/O 引脚与 VDD 和 VSS 间连有保护二极管。

图 5-3： GP2 框图



注 1： 在 I/O 引脚与 VDD 和 VSS 间连有保护二极管。

图 5-4： GP3 框图



注 1： GP3/MCLR 引脚只与 VSS 连有保护二极管。

 初稿

表 5-3： 端口寄存器汇总

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
N/A	TRISGPIO	—	—	—	—	I/O 控制寄存器				---- 1111	---- 1111
N/A	OPTION	$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1--1 1111	q--q 1111
03h	STATUS	GPWUF	—	—	TO	PD	Z	DC	C	0001 1xxx	qq0q quuu[(1)]
06h	GPIO	—	—	—	—	GP3	GP2	GP1	GP0	---- xxxx	---- uuuu

图注： 端口寄存器不使用阴影单元，读为 0， - = 未用，读为 0， x = 未知，u = 不变，
q = 取值视条件而定。

注 1： 如果复位是由电平变化唤醒引起的，那么 bit 7 = 1。所有其他复位， bit 7 = 0。

5.4 I/O 编程注意事项

5.4.1 双向 I/O 端口

某些指令在内部是先执行一个读操作，再执行一个写操作。例如 BCF 和 BSF 指令，这两个指令将整个端口的数据读入 CPU，执行位操作，然后重写结果。当一个端口既有输入引脚又有输出引脚时，对其执行指令必须加倍小心。例如，对 GPIO 的 bit 2 执行 BSF 操作时，先将 GPIO 的全部 8 位数据读入 CPU，然后将 bit 2 置 1，最后将 GPIO 的值写入输出锁存器。如果 GPIO 的另一个位被用作双向 I/O 引脚 （假设为 bit 0），并且该引脚被定义为输入，则该引脚当前的输入信号被读入 CPU，然后将值重新写入该特定引脚的数据锁存器，覆盖先前的内容。只要该引脚保持输入模式，就毫无问题。然而，如果 bit 0 随后变成了输出模式，则输出锁存器的内容将无法得知。

例 5-1 给出了对一个 I/O 端口执行两条连续的读—修改—写指令 （如 BCF 和 BSF 等）的效果。

当某引脚设置为输出时，不应通过外部器件来驱动该引脚为高电平或低电平，以达到改变该引脚电平（“线或”或 “线与”）的目的。否则，导致的高输出电流可能会损坏器件。

例 5-1： 对 I/O 端口执行读—修改—写指令

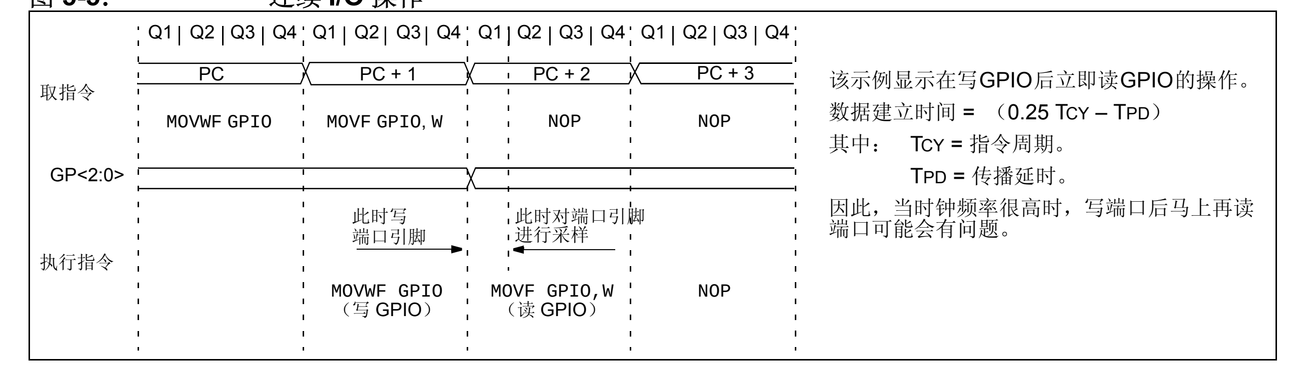
```
;Initial GPIO Settings
;GPIO<3:2> Inputs
;GPIO<1:0> Outputs
;
;                    GPIO latch   GPIO pins
;                    ----------   ----------
 BCF    GPIO,  1 ;---- pp01     ---- pp11
 BCF    GPIO,  0 ;---- pp10     ---- pp11
 MOVLW  007h;
 TRIS   GPIO     ;---- pp10     ---- pp11
;
```

注： 用户可能希望引脚值为 ---- pp00。第 2 条 BCF 指令将 GP1 锁存为该引脚的值（高电平）。

5.4.2 I/O 端口上的连续操作

对 I/O 端口的写操作实际发生在指令周期的末尾时刻，但对于读操作，在指令周期的开始处所读的数据就必须有效（见图 5-5）。因此，如果对同一个 I/O 端口执行写操作之后，接着执行读操作，就必须特别小心。指令执行的顺序应该是：等引脚电压达到稳定（与负载有关）后，才执行下一条指令将端口值读入 CPU。否则，读入的可能是引脚的前一个状态而不是新状态。当状态不确定时，最好用一条 NOP 指令或者其他不访问 I/O 端口的指令隔开这些指令。

图 5-5： 连续 I/O 操作



注：

6.0 TMR0 模块和 TMR0 寄存器

Timer0 模块具有如下特性：

- 8 位定时器 / 计数器寄存器 TMR0
  - 可读写
- 软件可编程的 8 位预分频器
- 内部或外部时钟选择
  - 外部时钟的边沿选择

图 6-1 是 Timer0 模块的简化框图。

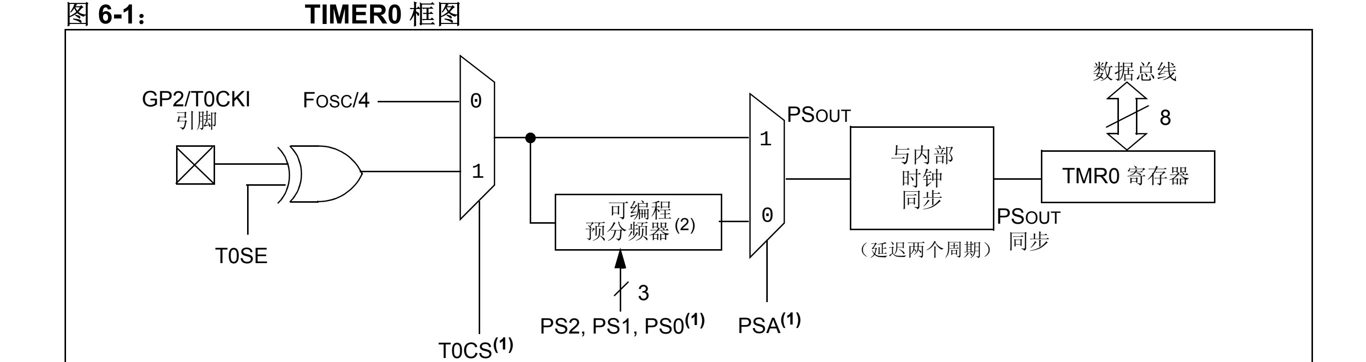
通过清零 T0CS 位 （OPTION<5>）选择定时器模式。在定时器模式下，Timer0 模块在每个指令周期都会递增（不使用预分频器）。如果对TMR0寄存器执行写操作，在接下来的两个指令周期禁止递增（图 6-2 和图 6-3）。用户可通过将校正值写入 TMR0 寄存器来避开这个问题。

通过将 T0CS 位 （OPTION<5>）置 1 来选择计数器模式。在此模式下， Timer0 会在 T0CKI 引脚的每个上升沿或下降沿递增。具体的计数沿由时钟源边沿选择位T0SE（OPTION<4>）决定。通过清零 T0SE 位来选择上升沿。在**第 6.1 节 "使用外部时钟作为 Timer0 的时钟源"**对外部时钟输入的限制进行了详细讨论。

Timer0 模块或看门狗定时器中只有一个可以使用预分频器，两者不能同时使用。预分频器的分配由控制位PSA（OPTION<3>）通过软件来控制。通过将 PSA 位清零将预分频器分配给 Timer0。不可直接读写该预分频器。如果将预分频器分配给 Timer0 模块，可供选择的预分频比有 1:2、1:4...，1:256。**第 6.2 节"预分频器"**详细介绍了预分频器的操作。

表 6-1 是与 Timer0 模块相关的寄存器的汇总。

图 6-1： TIMER0 框图



注 1： T0CS、 T0SE、 PSA、 PS2、 PS1 和 PS0 位于选项寄存器中。
2： Timer0 与看门狗定时器共用该预分频器 （图 6-5）。

图 6-2： TIMER0 时序图：内部时钟 / 无预分频器

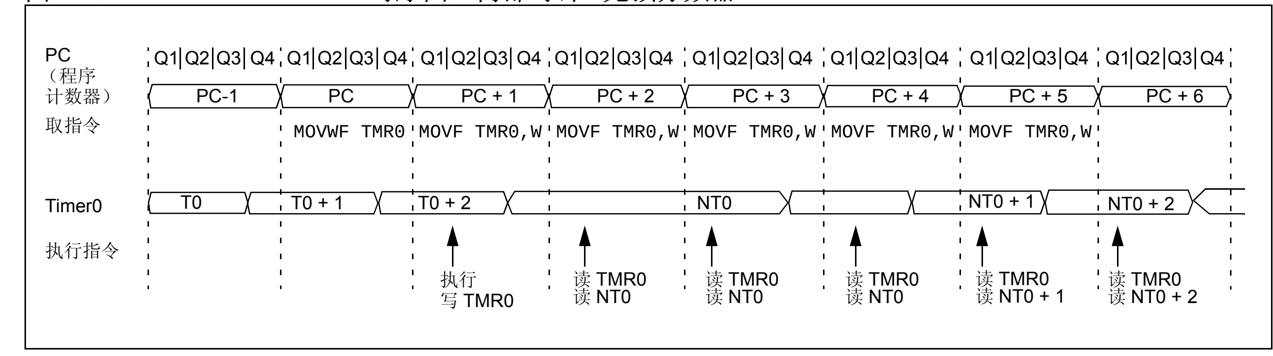


图 6-3： TIMER0 时序图：内部时钟 / 分频比为 1:2

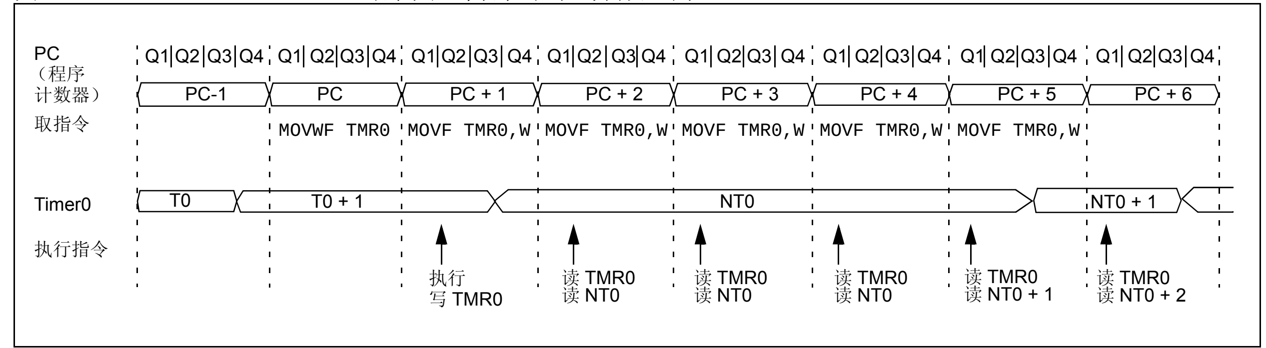


表 6-1： 与 TIMER0 相关的寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
01h	TMR0	Timer0——8 位实时时钟 / 计数器								xxxx xxxx	uuuu uuuu
N/A	OPTION	$\overline{GPWU}$	$\overline{GPPU}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	TRISGPIO(1)	—	—	—	—	I/O 控制寄存器				---- 1111	---- 1111

图注： Timer0 不使用阴影单元， - = 未用， x = 未知， u = 不变。
注 1： 当 T0CS = 1 时， T0CKI 引脚的 TRIS 位无效。

6.1 使用外部时钟作为 Timer0 的时钟源

为 Timer0 提供的外部时钟输入必须满足特定条件。要求外部时钟与内部相位时钟 （Tosc）同步。在同步之后，要经过一定延时， Timer0 才会递增。

6.1.1 外部时钟同步

如果不使用预分频器，那么外部时钟就是预分频器的输出。在内部时钟的 Q2 和 Q4 节拍对预分频器输出进行采样可实现 T0CKI 与内部相位时钟的同步 （图 6-4）。因此，要求 T0CKI 引脚信号的高电平时间至少为 2 个 Tosc（加上一小段 2Tt0H 的 RC 延时），并且低电平时间也至少为 2 个 Tosc （加上一小段 2Tt0H 的 RC 延时）。请参见所需器件的电气规范。

如果使用了预分频器，外部时钟输入要先经过异步纹波计数型预分频器的分频，从而使预分频器的输出对称。为了使外部时钟满足采样要求，必须考虑纹波计数器的影响。因此， T0CKI 的周期必须至少为 4 个 Tosc （加上一小段 4Tt0H 的 RC 延时）除以预分频值。T0CKI 引脚上的高低电平持续时间只须满足 Tt0H 的最小脉宽要求即可。请参见所需器件电气规范中的参数 40、 41 和 42。

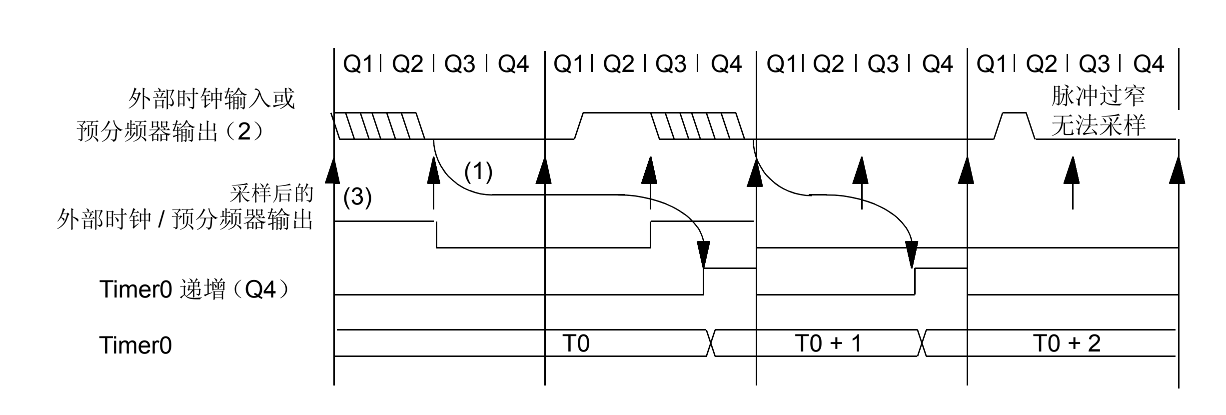
 初稿

6.1.2 TIMER0 递增延时

由于预分频器输出要与内部时钟同步，所以从出现外部时钟沿到 Timer0 模块递增会有一小段延时。图 6-4 说明了从出现外部时钟沿到定时器执行递增操作之间的延时。



图 6-4： TIMER0 和外部时钟的时序图



注 1：时钟输入发生变化到 Timer0 递增会有 3 到 7 个 TOSC（持续时间 Q = TOSC）的延时。因此，测量 Timer0 输入信号两个边沿之间的间隔时，其最大误差为 ± 4TOSC。

2：不选择预分频器时为外部时钟；否则为预分频器的输出。

3：箭头所指为采样时刻。

6.2 预分频器

8 位计数器可分别被用作 Timer0 模块的预分频器和看门狗定时器 （WDT）的后分频器 （见第 **8.6 节 “看门狗定时器 （WDT）”**）。为简化起见，在本数据手册中将它们统称为预分频器。

注：	Timer0 模块或看门狗定时器中只有一个可以使用预分频器，两者不能同时使用。 因此，如果预分频器分配给 Timer0 模块，看门狗定时器就没有预分频器可用了，反之亦然。

PSA 和 PS<2:0> 位（OPTION<3:0>）确定预分频器的分配和预分频比。

如果分配给 Timer0 模块，所有写入 TMR0 寄存器的指令（如 `CLRF 1`、`MOVWF 1`、`BSF 1`、`x` 等）都会使预分频器清零。如果分配给看门狗定时器，`CLRWDT` 指令将使该预分频器和看门狗定时器同时清零。不可直接读写该预分频器。复位时，预分频器内容为全 `0`。

6.2.1 切换预分频器的分配

预分频器的分配完全由软件控制，即可以在程序执行期间“随时”更改。为避免器件意外复位，请在将预分频器从分配给 Timer0 切换为分配给 WDT 时，务必执行以下指令序列 （例 6-1）。

例 6-1： 切换预分频器 （TIMER0→ WDT）

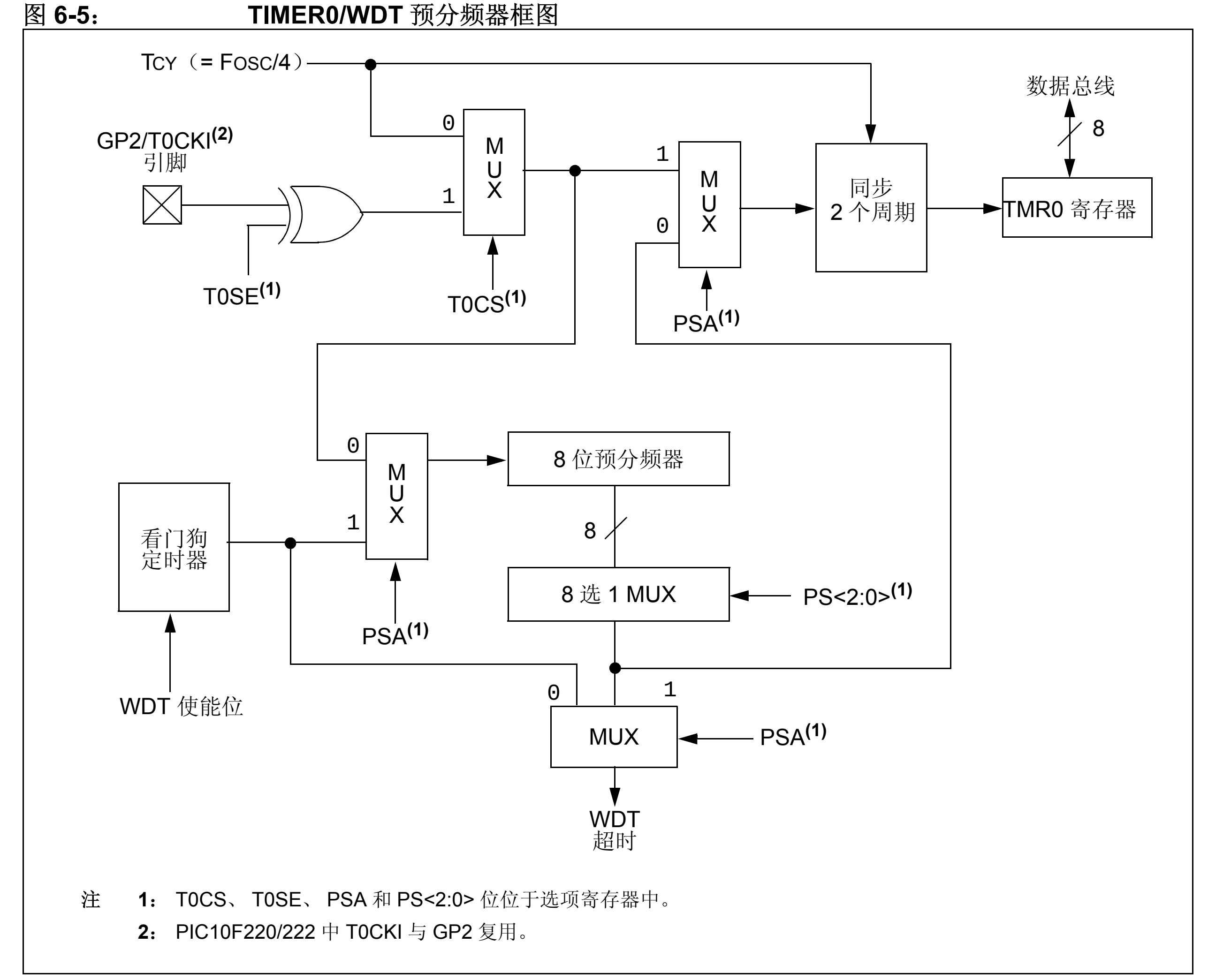
```
CLRWDT                ;Clear WDT
CLRF    TMR0          ;Clear TMR0 & Prescaler
MOVLW   '00xx1111'b   ;These 3 lines (5, 6, 7)
OPTION                ;are required only if
                      ;desired
CLRWDT                ;PS<2:0> are 000 or 001
MOVLW   '00xx1xxx'b   ;Set Postscaler to
OPTION                ;desired WDT rate
```

要将预分频器从分配给 WDT 切换为分配给 Timer0 模块，请执行例 6-2 中给出的序列。即便 WDT 被禁用也要执行该序列。切换预分频器前要执行 CLRWDT 指令。

例 6-2： 切换预分频器 （WDT→TIMER0）

```
CLRWDT                ;Clear WDT and
                      ;prescaler
MOVLW   'xxxx0xxx'    ;Select TMR0, new
                      ;prescale value and
                      ;clock source
OPTION
```

图 6-5： TIMER0/WDT 预分频器框图



注 1： T0CS、 T0SE、 PSA 和 PS<2:0> 位位于选项寄存器中。
2： PIC10F220/222 中 T0CKI 与 GP2 复用。

 初稿

7.0 模数 （A/D）转换器

A/D 转换器可以将模拟信号转换为表示该信号的 8 位数字信号。

7.1 时钟分频器

A/D 转换器仅有一个 INTOSC/4 时钟源设置。ADC 转换需要 13 个 T_{AD} 周期。分频值不会影响执行转换需要的 T_{AD} 周期数，它只决定 T_{AD} 周期的长度。

注：	由于时钟分频值固定，转换会在 13 个 CPU 指令周期内完成。

7.2 参考电压

A/D 转换器没有外部参考电压。 A/D 转换器参考电压始终为 V_{DD}。

7.3 模拟模式选择

ANS<1:0> 位用于将引脚配置为模拟输入。发生任何复位时，ANS<1:0> 都默认为 11，从而将 AN0 和 AN1 引脚配置为模拟输入。配置为模拟输入的引脚不可用于数字输出。在进行转换时，用户不应改变 ANS 位。不论 ADON 的状态如何， ANS 位始终有效。

7.4 A/D 转换器通道选择

CHS 位用于选择 A/D 转换器采样的模拟通道。用户不能在转换中改变 CHS 位的值。要采集一个模拟信号，CHS 选择必须与 ANS 位选定的一个引脚匹配。不管 ANS 位状态如何，始终可以选择内部绝对参考电压。当器件进入休眠模式时，将丢失所有有关通道选择的信息。

注：	当 ADON 置 1 时，即使没有选择将任何通道作为模拟输入， A/D 转换器模块仍会消耗功率。对于低功耗应用，建议在不使用 A/D 转换器时将 ADON 位清零。

7.5 GO/$\overline{\text{DONE}}$ 位

GO/$\overline{\text{DONE}}$ 位用于确定转换状态、启动一个转换并手动停止正在进行的转换。将 GO/$\overline{\text{DONE}}$ 位置 1 将启动转换。当转换完成后，ADC 模块将 GO/$\overline{\text{DONE}}$ 位清零。手动将 GO/$\overline{\text{DONE}}$ 位清零可终止正在进行的转换。手动终止转换将导致 ADRES 中只有部分转换结果。

当器件进入休眠模式时， GO/$\overline{\text{DONE}}$ 位被清零，停止当前转换。ADC 没有专用的振荡器，它使用的是系统时钟。

当 ADON 清零时，不能将 GO/$\overline{\text{DONE}}$ 位置 1。

7.6 休眠模式

ADC 没有专用的 ADC 时钟，因此在休眠模式不可能工作。如果在转换进行过程中执行一条 SLEEP 指令，则 GO/$\overline{\text{DONE}}$ 和 ADON 位将被清零。这样会停止转换，并且ADC模块将关闭以节约功耗。由于转换过程的特性，ADRES 可能保存部分转换的结果。只有在进入休眠模式前至少转换 1 位数据， ADRES 中才有部分完成的转换数据。CHS 位复位到默认状态，即 CHS<1:0> = 11。

要得到正确的转换结果， T_{AD} 必须符合以下条件：

- 500 ns < T_{AD} < 50 µs
- T_{AD} = 1/(F_{OSC}/ 分频值)

表 7-1： 休眠和唤醒对 ADCON0 的影响

	ANS1	ANS0	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON
休眠前	x	x	x	x	0	0
休眠前	x	x	x	x	1	1
进入休眠	不变	不变	1	1	0	0
唤醒	1	1	1	1	0	0

7.7 模拟转换结果寄存器

ADRES 寄存器保存上次转换的结果。这些结果保存到下一次模拟转换的采样阶段。当采样周期结束后，ADRES 清零（= 0）。“前导位（Leading One）”向右移入 ADRES 中作为内部转换完成位。从高位开始按位依次进行转换，“前导位”不断右移以使后来转换的位填入 ADRES 中。当“前导位”右移 9 次后转换完成；“前导位”移出且 GO/DONE 位清零。

如果在转换过程中 GO/DONE 位由软件清零，转换即停止。 ADRES 中的数据为部分转换结果。此数据对于已转换的位是有效的。“前导位”的位置决定已转换位的个数。在 GO/DONE 位清零以前没有被转换的位是不可恢复的。

7.8 内部绝对参考电压

内部绝对参考电压的功能是在器件 VDD 范围内为转换提供一个恒定电压。A/D 转换与转换参考电压 VDD 成正比。转换 0.6V（典型值）恒定电压的结果取决于施加到器件 VDD 引脚上的电压。在 VDD 范围内参考电压的转换结果大约为：转换结果 = 0.6V/(VDD/256)

注： 绝对参考电压的实际值随温度和器件变化。转换还受 VDD 引脚上模拟噪声和 I/O 引脚上灌 / 拉电流产生的噪声的影响。

寄存器 7-1： ADCON0 寄存器

R/W-1	R/W-1	U-0	U-0	R/W-1	R/W-1	R/W-0	R/W-0
ANS1	ANS0[1,2]	—	—	CHS1[3]	CHS0[3]	GO/DONE[4]	ADON
bit 7							bit 0

bit 7 **ANS1**：ADC 模拟输入引脚选择位
1 = GP1/AN1 配置为模拟输入
0 = GP1/AN1 配置为数字 I/O

bit 6 **ANS0**：ADC 模拟输入引脚选择位 [1,2]
1 = GP0/AN0 配置为模拟输入
0 = GP0/AN0 配置为数字 I/O

bit 5-4 未用：读为 0

bit 3-2 **CHS<1:0>**：ADC 通道选择位 [3]
00 = 通道 00（GP0/AN0）
01 = 通道 01（GP1/AN1）
1X = 0.6V 绝对参考电压

bit 1 **GO/DONE**：ADC 转换状态位 [4]
1 = ADC 正在转换。将该位置 1 会启动 ADC 转换。当 ADC 转换完成，该位由硬件自动清零。
0 = ADC 转换完成 / 不在进行转换。在转换中手动将该位清零可中止当前转换。

bit 0 **ADON**：ADC 使能位
1 = ADC 模块正在工作
0 = ADC 被关闭且不消耗功耗

注 **1：** 当 ANS 位置 1 时，不管先前将引脚定义为何功能，都会自动强制选定通道为模拟模式。
2： 不管 ADON 状态如何， ANS<1:0> 始终有效。
3： 复位后 CHS<1:0> 位默认为 11。
4： 如果 ADON 位清零， GO/DONE 位就不能被置 1。

图注：			
R = 可读位	W = 可写位	U = 未用位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

寄存器 7-2： ADRES 寄存器

R-X	R-X	R-X	R-X	R-X	R-X	R-X	R-X
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7							bit 0

bit 7-0 **PC<7:0>**

图注：			
R = 可读位	W = 可写位	U = 未用位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

注：

8.0 CPU 的特殊性能

单片机与其他处理器的区别在于其具有处理实时应用所需的特殊电路。PIC10F220/222 单片机具有许多特性，旨在最大限度地提高系统的可靠性，通过减少外部元件将成本降至最低，并且还提供了低功耗工作模式和代码保护功能。这些特性如下：

- 复位：
  - 上电复位 （POR）
  - 器件复位定时器 （DRT）
  - 看门狗定时器 （WDT）
  - 引脚电平变化时从休眠模式唤醒
  - 比较器变化时从休眠模式唤醒
- 休眠模式
- 代码保护
- ID 单元
- 在线串行编程功能
- 时钟输出

PIC10F220/222 器件具有一个看门狗定时器，只可以通过配置位 WDTE 来关闭它。为了增强可靠性，它依靠自己的 RC 振荡器运行。当使用 DRT 时，只有在 V_{DD} 上电复位时才有 1.125 ms （典型值）的延时。有了此片上定时器，大部分应用都不再需要外部复位电路。

休眠模式是为提供电流非常低的掉电模式而设计的。用户可以通过输入引脚或比较器的电平变化或者看门狗定时器超时溢出使器件从休眠模式唤醒。

8.1 配置位

PIC10F220/222 配置字由 12 位组成。可以编程配置位来选择不同的器件配置。一个位是看门狗定时器使能位，一个位是 MCLR 使能位，还有一个位用于代码保护（见寄存器 8-1）。

寄存器 8-1： PIC10F220/222 配置字 (1)

—	—	—	—	—	—	—	MCLRE	$\overline{CP}$	WDTE	$\overline{MCPU}$	IOSCFS

bit 11   bit 0

bit 11-5 未用：读为 0

bit 4 **MCLRE**：GP3/$\overline{MCLR}$ 引脚功能选择位
1 = GP3/$\overline{MCLR}$ 引脚功能是 $\overline{MCLR}$
0 = GP3/MCLR 引脚功能是数字 I/O， $\overline{MCLR}$ 在内部与 V_{DD} 相连

bit 3 **$\overline{CP}$**：代码保护位
1 = 关闭代码保护
0 = 打开代码保护

bit 2 **WDTE**：看门狗定时器使能位
1 = 使能 WDT
0 = 禁止 WDT

bit 1 **$\overline{MCPU}$**：主复位上拉使能位
1 = 禁止上拉
0 = 使能上拉

bit 0 **IOSCFS**：内部振荡频率选择位 (2)
1 = 8 MHz
0 = 4 MHz

注 **1**：如需确定访问配置字的方法，请参见 “*PIC10F220/222 Memory Programming Specification*” （DS41266）。在器件工作过程中用户不能寻址配置字寄存器。

2：要使能频率选择， MCLRE 必须置 1。

图注：			
R = 可读位	W = 可写位	U = 未用位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

8.2 振荡器配置

8.2.1 振荡器类型

PIC10F220/222 器件仅具有内部振荡器模式。

- INTOSC：内部 4 MHz/8 MHz 振荡器

8.2.2 内部 4 MHz/8 MHz 振荡器

内部振荡器提供 4 MHz/8 MHz（标称值）系统时钟（见**第 10.0 节“电气规范”**了解频率随电压和温度变化的差异）。

此外，校准指令被编写在存储器的最后一个地址单元中，其中包括内部振荡器的校准值。不管是否设置了代码保护，此单元总是没有代码保护的。此值被编程为 `MOVLW XX` 指令，其中 XX 是校准值，被放在复位向量处。这将在复位时把校准值载入 W 寄存器，然后 PC 将返回至地址 0x000 处的用户程序。然后用户可以选择是将此值写入 OSCCAL 寄存器（05h）还是忽略它。

当将校准值写入 OSCCAL 寄存器时，将“调整”内部振荡器以消除由于制造工艺引起的振荡频率差异。

注：	擦除器件同时也会擦除内部振荡器预编程的内部校准值。必须在擦除器件之前读取校准值，从而以后可以正确地重新编程。

8.3 复位

器件因不同类型的复位而有所区别：

- 上电复位（POR）
- 正常工作时的 $\overline{\text{MCLR}}$ 复位
- 休眠时的 $\overline{\text{MCLR}}$ 复位
- 正常工作时的 WDT 超时溢出复位
- 休眠时的 WDT 超时溢出复位
- 在引脚电平发生变化时从休眠模式唤醒

某些寄存器没有任何形式的复位，在上电复位时它们的状态是未知的，在任何其他复位时它们不变。大部分其他寄存器都会在正常工作过程中发生上电复位（POR）、MCLR、WDT 复位或引脚电平变化唤醒复位时复位到“复位状态”。它们在休眠过程中不受到 WDT 复位或 MCLR 复位的影响，因为这些复位被视为恢复正常运行。但 $\overline{\text{TO}}$、$\overline{\text{PD}}$ 和 GPWUF 位例外。在不同的复位情况下它们的置 1 和清零状况不同。在软件中使用这些位来确定复位的特性。所有寄存器的复位状态的完整说明 请参见表 8-1。

表 8-1： 寄存器的复位状态—— PIC10F220/222

寄存器	地址	上电复位	$\overline{\text{MCLR}}$ 复位、WDT 超时溢出、在引脚电平变化时唤醒
W	—	qqqq qqqu(1)	qqqq qqqu(1)
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu
PC	02h	1111 1111	1111 1111
STATUS	03h	0--1 1xxx	q00q quuu(2)
FSR	04h	111x xxxx	111u uuuu
OSCCAL	05h	1111 1110	uuuu uuuu
GPIO	06h	---- xxxx	---- uuuu
ADCON0	07h	11-- 1100	11-- 1100
ADRES	08h	xxxx xxxx	uuuu uuuu
OPTION	—	1111 1111	1111 1111
TRIS	—	---- 1111	---- 1111

图注： u = 不变， x = 未知， — = 未用位，读为 0， q = 取值视条件而定。
注 1： W 寄存器的 bit <7:2> 包含振荡器校准值，这是因为存储器顶部有一条 MOVLW XX 指令。
2： 请参见表 8-2 了解特定条件下的复位值。

表 8-2： 特殊寄存器的复位状态

	状态寄存器地址：03h	PCL 地址：02h
上电复位	0--1 1xxx	1111 1111
正常工作状态下的 $\overline{\text{MCLR}}$ 复位	0--u uuuu	1111 1111
休眠状态下的 $\overline{\text{MCLR}}$ 复位	0--1 0uuu	1111 1111
休眠状态下的 WDT 复位	0--0 0uuu	1111 1111
正常工作状态下的 WDT 复位	0--0 uuuu	1111 1111
引脚电平变化时从休眠模式唤醒	1--1 0uuu	1111 1111

图注： u = 不变， x = 未知

8.3.1 $\overline{\text{MCLR}}$ 使能

此配置位在未编程时（状态为 1）会使能外部 $\overline{\text{MCLR}}$ 功能引脚。在已编程时， $\overline{\text{MCLR}}$ 功能引脚将连接到内部 V_{DD} 并且分配该引脚为 I/O 引脚。请参见图 8-1。

图 8-1： $\overline{\text{MCLR}}$ 选择

8.4 上电复位 （POR）

PIC10F220/222 器件采用了一个片上上电复位（POR）电路，此电路为大部分上电情况提供内部芯片复位。

片上上电复位电路将芯片保持在复位状态，直到 V_{DD} 上升到足以正常工作的电平。要利用内部上电复位，可以将 GP3/$\overline{\text{MCLR}}$/V_{PP} 引脚编程为 $\overline{\text{MCLR}}$ 引脚，并通过一个电阻将此引脚连接到 V_{DD}，或者将此引脚编程为 GP3。内部弱上拉电阻通过晶体管实现（上拉电阻范围请参见表 10-2）。这将省掉产生上电复位通常所需要的外部 RC 组件。指定了 V_{DD} 的最大上升时间。如需详细信息，请参见**第 10.0 节 "电气规范"**。

当器件开始正常工作时 （退出复位条件），器件工作参数（电压、频率和温度等）必须符合一定的条件以保证正常工作。如果不符合这些条件，器件必须被保持在复位状态，直到符合工作参数条件为止。

图 8-2 给出了片上上电复位电路的的简化框图。

上电复位电路和器件复位定时器（见**第 8.5 节 "器件复位定时器 （DRT）"**）电路是紧密相关的。上电时，复位锁存器置 1 且 DRT 被复位。DRT 定时器一旦检测到 $\overline{\text{MCLR}}$ 引脚为高电平就开始计数。在延时周期（通常为 1.125 ms）之后，它将使复位锁存器复位，从而结束片上复位信号。

图 8-3 所示为一个 $\overline{\text{MCLR}}$ 引脚保持为低电平的上电示例。在 $\overline{\text{MCLR}}$ 引脚变为高电平之前，允许 V_{DD} 上升并保持稳定。芯片实际上会在 $\overline{\text{MCLR}}$引脚变为高电平之后的 T_{DRT} ms 之后，退出复位状态。

在图 8-4 中，使用片上上电复位功能，$\overline{\text{MCLR}}$ 引脚和 V_{DD} 被连接在一起或者 $\overline{\text{MCLR}}$引脚被编程为 GP3。V_{DD} 在起振定时器超时溢出前趋于稳定并且能够正常复位。但是，图 8-5 描述了一个当 V_{DD} 上升过于缓慢时出现的问题。从 DRT 检测到 $\overline{\text{MCLR}}$ 引脚为高电平到 $\overline{\text{MCLR}}$ 和 V_{DD} 实际上达到其最大值之间的时间过长。在这种情况下，当起振定时器延时结束时，V_{DD} 尚未达到 V_{DD} 最小值，芯片可能无法正常运行。如果发生这种情况，建议使用外部 RC 电路以延长上电复位延时 （图 8-4）。

注：	当器件开始正常工作时（退出复位状态），器件工作参数 （电压、频率和温度等）必须符合一定的条件以保证正常工作。如果不符合这些条件，那么器件必须保持在复位状态，直到符合工作条件为止。

要了解更多信息，请参见应用笔记 *AN522 "Power-Up Considerations"*（DS00522）和 *AN607 "Power-up Trouble Shooting"*（DS00607）。

图 8-2： 片上复位电路的简化框图

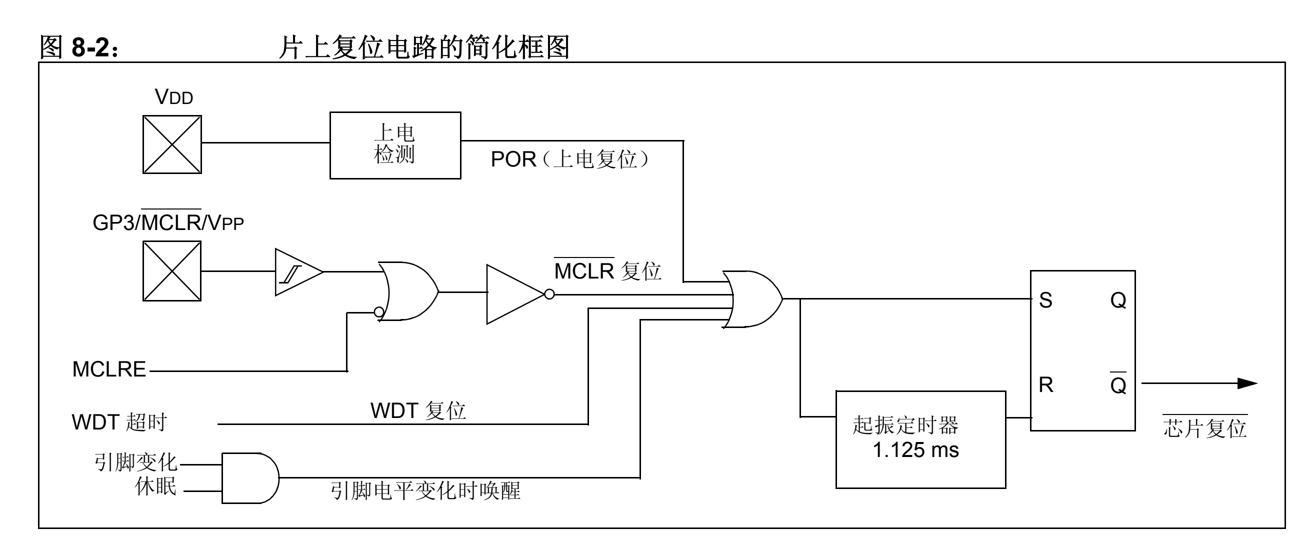


图 8-3： 上电延时时序 （**MCLR** 拉为低电平）

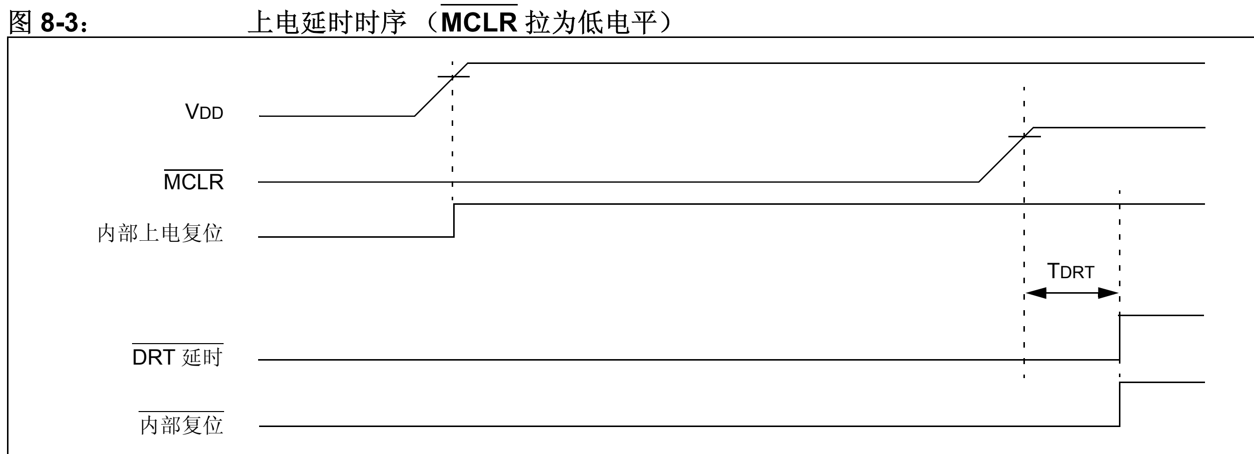
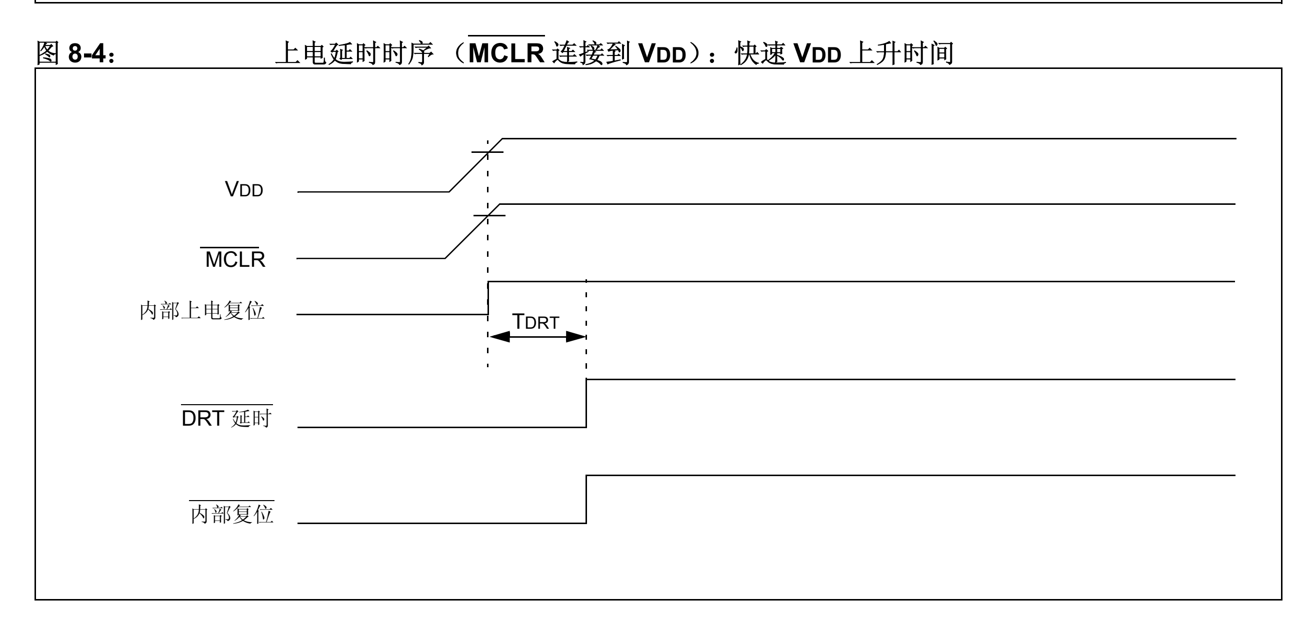
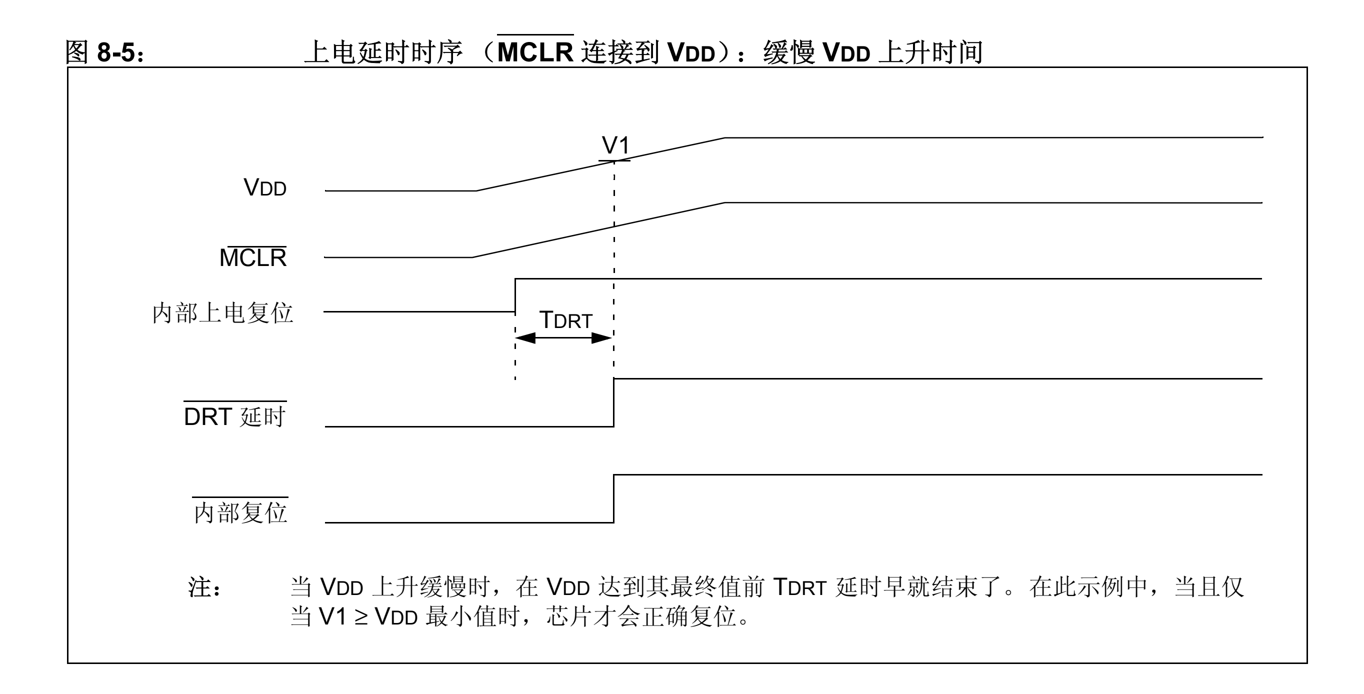


图 8-4： 上电延时时序 （**MCLR** 连接到 **VDD**）：快速 **VDD** 上升时间



 初稿

图 8-5：　　上电延时时序 （$\overline{\text{MCLR}}$ 连接到 VDD）：缓慢 VDD 上升时间



注：　当 VDD 上升缓慢时，在 VDD 达到其最终值前 TDRT 延时早就结束了。在此示例中，当且仅当 V1 ≥ VDD 最小值时，芯片才会正确复位。

8.5 器件复位定时器 （DRT）

在 PIC10F220/222 器件上，每次当器件上电时 DRT 都会运行。

DRT 用内部振荡器作为时钟来工作。只要 DRT 是活动的，处理器就保持在复位状态。DRT 延时使 VDD 上升到 VDD 最小值以上，并使振荡器达到稳定。

在 MCLR 达到逻辑高电平（VIH MCLR）之后，片上 DRT 会将器件保持在复位状态大约 1.125 ms。在大多数情况下，不要求将 GP3/MCLR/VPP 编程为 MCLR 或使用外部 RC 网络连接到 MCLR 输入引脚。这可以在对成本和 / 或空间控制严格的应用中节省开支和空间，并且允许将 GP3/MCLR/VPP 引脚用作通用输入引脚。

由于 VDD、温度和制造工艺的不同，芯片之间的器件复位延时也各不相同。详情请参见交流参数。

复位源为上电复位、MCLR、WDT 超时溢出和在引脚电平变化时唤醒。请参见 **第 8.9.2 节 “从休眠唤醒” 中的注 1**、 **2** 和 **3**。

表 8-3： DRT （器件复位定时器周期）

振荡器	上电复位	其他复位
INTOSC	1.125 ms（典型值）	10 μs（典型值）

8.6 看门狗定时器 （WDT）

看门狗定时器（WDT）是自由运行的片上 RC 振荡器，它不需要任何外部组件。此 RC 振荡器独立于内部 4 MHz/8 MHz 振荡器。这意味着即使主处理器时钟已经停止（如通过执行 SLEEP 指令）WDT 将仍然运行。在正常工作或休眠过程中， WDT 复位或唤醒复位都会产生器件复位。

TO 位（STATUS<4>）将在看门狗定时器复位时清零。

可以通过将配置位 WDTE 编程为 0 来永久禁止 WDT（见第 **8.1** 节 **“配置位”**）。如需确定访问配置字的方法，请参见 PIC10F220/222 编程规范。

8.6.1 WDT 周期

WDT 的正常超时溢出周期为 18 ms（没有预分频器）。如果需要更长的超时溢出周期，可以通过写选项寄存器为 WDT（在软件控制下）分配一个分频比最高为 1:128 的预分频器。因此，可以实现一个正常的 2.3 秒的超时溢出周期。此周期根据温度、VDD 以及各器件的不同制造工艺而有所不同 （见直流规范）。

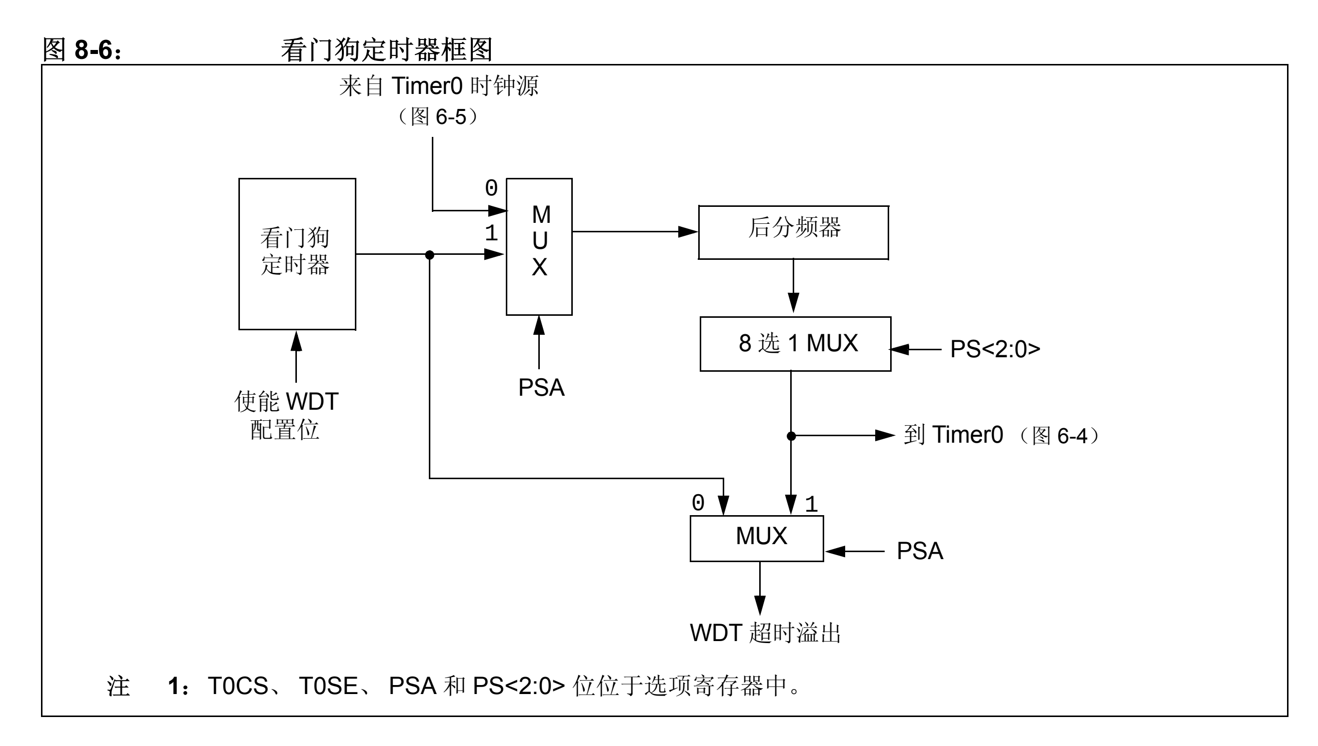
在最坏的情况下 （VDD = 最小值、温度 = 最大值并且 WDT 预分频比最大），在发生 WDT 超时溢出之前会有几秒钟的延迟。

8.6.2 WDT 编程注意事项

CLRWDT 指令将清零 WDT 和后分频器 （如果将后分频器分配给 WDT），并阻止它超时溢出和产生器件复位。

SLEEP 指令将复位 WDT 和后分频器（如果将后分频器分配给 WDT）。这将在 WDT 唤醒复位之前提供一个最大休眠时间。

图 8-6： 看门狗定时器框图



注 **1：** T0CS、 T0SE、 PSA 和 PS<2:0> 位位于选项寄存器中。

表 8-4： 与看门狗定时器相关的寄存器汇总

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
N/A	OPTION	GPWU	GPPU	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

图注： 阴影单元 = 看门狗定时器不使用

8.7 超时时序、掉电和从休眠唤醒状态位（TO/PD/GPWUF/CWUF）

可以通过测试状态寄存器中的 TO、PD 和 GPWUF 位来确定是否上电条件、MCLR、看门狗定时器（WDT）复位或引脚或比较器电平变化唤醒已经导致复位条件产生。

表 8-5： 复位后 TO/PD/GPWUF 的状态

GPWUF	TO	PD	复位原因
0	0	0	WDT 从休眠模式唤醒
0	0	u	WDT 超时溢出 （非休眠模式）
0	1	0	MCLR 从休眠模式唤醒
0	1	1	上电
0	u	u	在非休眠模式下发生 MCLR 复位
1	1	0	在引脚电平发生变化时从休眠模式唤醒

图注： u = 不变

注 **1：** TO、PD 和 GPWUF 位保持其状态（u），直到发生复位。MCLR 输入引脚上的低脉冲不改变 TO、PD 或 GPWUF 状态位。

8.8 欠压复位

欠压是器件电压（VDD）降到最小值以下（不为零）后又回升的状态。欠压时器件应该复位。

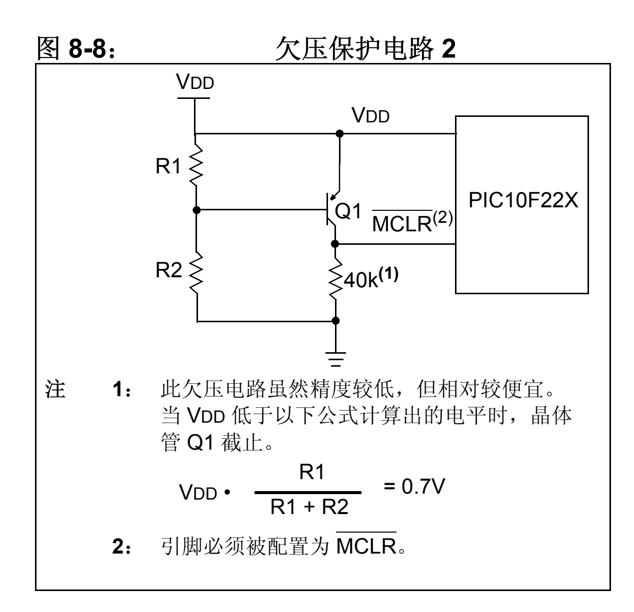
要在欠压时复位 PIC10F220/222 器件，需要建立外部欠压保护电路，如图 8-7 和图 8-8 所示。

图 8-7： 欠压保护电路 1

注 **1：** 当 VDD 低于 Vz + 0.7V （其中 Vz = 齐纳电压）时，此电路将激活复位。

2： 引脚必须被配置为 $\overline{\text{MCLR}}$。

图 8-8： 欠压保护电路 2

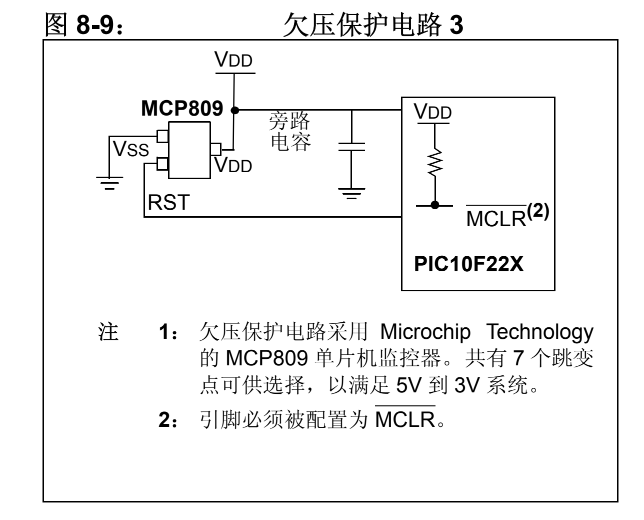


注 **1：** 此欠压电路虽然精度较低，但相对较便宜。当 VDD 低于以下公式计算出的电平时，晶体管 Q1 截止。

$$V_{DD} \cdot \frac{R1}{R1 + R2} = 0.7V$$

2： 引脚必须被配置为 $\overline{\text{MCLR}}$。

图 8-9： 欠压保护电路 3



注 **1：** 欠压保护电路采用 Microchip Technology 的 MCP809 单片机监控器。共有 7 个跳变点可供选择，以满足 5V 到 3V 系统。

2： 引脚必须被配置为 $\overline{\text{MCLR}}$。

8.9 掉电模式 （休眠）

器件可以掉电（休眠）然后再上电（从休眠模式唤醒）。

8.9.1 休眠

通过执行 SLEEP 指令进入掉电模式。

使能时，看门狗定时器将被清零但仍然保持运行，$\overline{\text{TO}}$ 位（STATUS<4>）被置 1，$\overline{\text{PD}}$ 位（STATUS<3>）被清零并且振荡器驱动器被关闭。I/O 端口保持 SLEEP 指令执行前的状态（驱动为高电平、驱动为低电平或高阻态）。

注：	WDT 超时溢出所产生的复位不会将 $\overline{\text{MCLR}}$ 引脚驱动为低电平。

为了达到掉电时的最低电流消耗，T0CKI 输入电平应该为 VDD 或 VSS，而且在 $\overline{\text{MCLR}}$ 使能时， GP3/$\overline{\text{MCLR}}$/VPP 引脚必须处于逻辑高电平。

8.9.2 从休眠唤醒

器件可以通过以下事件之一从休眠模式唤醒：

1. 当配置为 MCLR 时，GP3/MCLR/VPP 引脚上发生外部复位输入。
2. 看门狗定时器超时溢出复位（如果 WDT 使能）。
3. 当使能了电平变化唤醒时，输入引脚 GP0、GP1，或 GP3 上发生电平变化。

这些事件都将导致器件复位。可以使用 TO、PD 和 GPWUF 位来确定器件复位的原因。如果发生 WDT 超时溢出（并引起唤醒），TO 位被清零。PD 位在上电时置 1，在调用 SLEEP 时清零。GPWUF 位表示休眠时 GP0、GP1 或 GP3 引脚状态的变化（从 GP 端口上最后一次寄存器或位操作之后）。

注：	**提醒：**请在进入休眠模式之前读取输入引脚。当器件处于休眠模式时，引脚上的值从上次读该引脚时的的状态发生改变时，就会发生唤醒。如果发生电平变化时唤醒，并且引脚在进入休眠模式前未被读取，即使在休眠模式中没有引脚发生变化，也会立即发生唤醒。

注：	无论是什么原因引起唤醒，当器件从休眠模式唤醒时， WDT 都将被清零。

8.10 程序验证和代码保护

如果代码保护位未被编程，验证时可以读出片上程序存储器。

不管代码保护位的设置如何，都可以读前 64 个单元和最后一个单元 （复位向量）。

8.11 ID 单元

有四个存储单元指定为ID单元，用户可将校验和与其他代码标识号存储其中。在正常执行过程中这些单元不可访问，但是在编程 / 校验过程中这些单元可读写。

仅使用 ID 单元的低 4 位并总是将高 8 位编程为 0。

8.12 在线串行编程

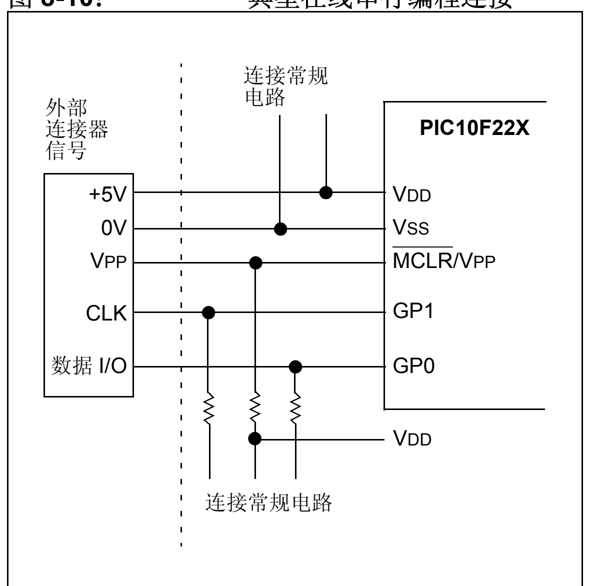
PIC10F220/222 单片机可在最终应用电路中被串行编程。只需要 5 根线即可完成这一操作，其中时钟线、数据线各一根，其余 3 根分别是电源线、接地线和编程电压线。这使用户可使用未编程的器件制造电路板，而仅在产品交付前才对单片机进行编程。这样还可以将最新的固件或定制的固件编程到器件中。

通过将 MCLR（VPP）引脚的电平从 VIL 拉升到 VIHH，同时将 GP1 和 GP0 引脚的电平保持为低电平可将器件保持在编程 / 校验模式（见编程规范）。此时 GP1 成为编程时钟而 GP0 成为编程数据。在此模式中 GP1 和 GP0 都是施密特触发器输入。

复位之后，为器件提供一条 6 位的命令。根据命令，可以为器件提供 16 位程序数据或者由器件提供 16 位程序数据，这取决于命令是一条载入命令还是一条读取命令。欲知串行编程的完整细节，请参见 PIC10F220/222 编程规范。

图 8-10 所示为典型的在线串行编程连接。

图 8-10： 典型在线串行编程连接



注：

9.0 指令集综述

PIC16 指令集具有高度正交性，分为以下三种基本类型：

- **字节**操作类指令
- **位**操作类指令
- **立即数和控制**操作类指令

每条 PIC16 指令都是 12 位字的，由**操作码**（指定指令类型）和一个或多个**操作数**（指定指令操作）组成。图 9-1 中显示了每种指令类型的格式，而表 9-1 总结了各种操作码字段。

对于**字节操作**指令，“f”为代表文件寄存器的指示符，而“d”为代表目标寄存器的指示符。文件寄存器指示符指定指令将会使用哪一个文件寄存器。

目标寄存器指示符指定操作结果的存放位置。如果“d”为 0，结果存放在 W 寄存器中。如果“d”为 1，结果存放在指令指定的文件寄存器中。

对于**位操作类**指令，“b”为代表位域的指示符，用于选择操作所影响的位，而“f”则代表相应位所在的寄存器的地址。

对于**立即数和控制**操作类指令，“k”代表一个 8 位或 9 位常数或立即数值。

表 9-1： 操作码字段说明

字段	说明
f	文件寄存器地址（0x00 到 0x7F）
W	工作寄存器（累加器）
b	8 位文件寄存器内的位地址
k	立即数字段、常数或标号
x	与取值无关的单元（= 0 或 1）。 汇编器将生成 x = 0 的代码。为了与所有的 Microchip 软件工具兼容，建议使用这种形式。
d	目标选择； d = 0（将结果保存在 W 寄存器中） d = 1（将结果保存在文件寄存器 f 中） 默认 d = 1
label	标号名
TOS	栈顶
PC	程序计数器
WDT	看门狗定时器计数器
$\overline{TO}$	超时溢出位
$\overline{PD}$	掉电位
dest	目标寄存器，可以是 W 寄存器也可以是文件寄存器
[ ]	选项
( )	内容
→	赋值给
< >	寄存器位域
∈	属于指定的集合
斜体	用户定义项（字体为 courier）

所有指令都在一个指令周期内执行，除非条件测试为真或者指令执行改变了程序计数器的值。当上述特殊情况发生时，指令的执行就需要两个指令周期。每个指令周期由 4 个振荡周期组成。因此，对于频率为 4 MHz 的振荡器，其正常的指令执行时间为 1 µs。如果条件测试为真或者指令执行改变了程序计数器的值，那么指令的执行时间将为 2 µs。

图 9-1 显示指令可具有的 3 种一般格式。图中的所有示例都使用以下格式来表示一个 16 进制数：

0xhhh

其中 h 代表一个 16 进制数字。

图 9-1： 指令的一般格式

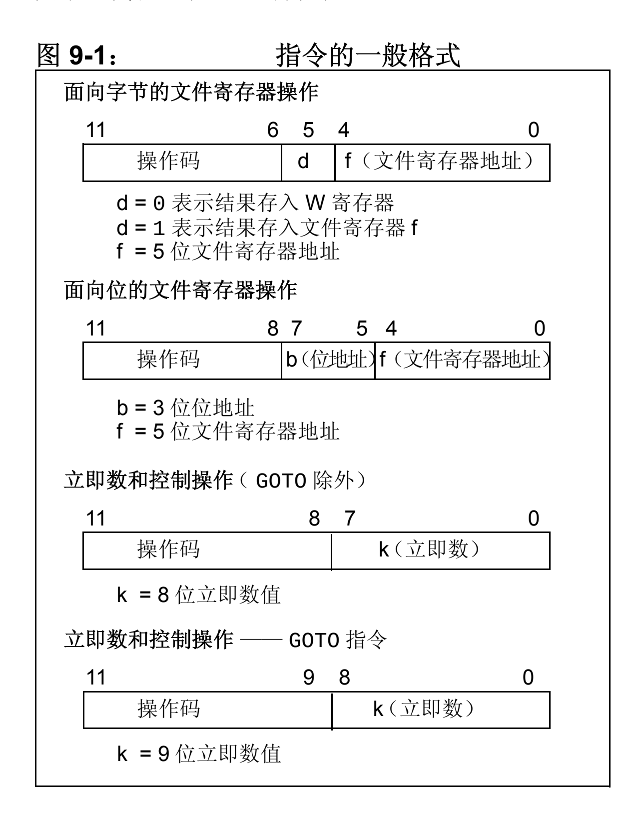


表 9-2： 指令集汇总

助记符，操作数		说明	周期数	12 位操作码			受影响的状态位	注
				MSb		LSb		
ADDWF	f,d	W和f相加	1	0001	11df	ffff	C、DC和Z	1,2,4
ANDWF	f,d	W和f作逻辑与运算	1	0001	01df	ffff	Z	2,4
CLRF	f	将f清零	1	0000	011f	ffff	Z	4
CLRW	—	将W寄存器清零	1	0000	0100	0000	Z	
COMF	f, d	f取反	1	0010	01df	ffff	Z	
DECF	f, d	f减1	1	0000	11df	ffff	Z	2,4
DECFSZ	f, d	f减1，为0则跳过	1	0010	11df	ffff	无	2,4
INCF	f, d	f增1	1	0010	10df	ffff	Z	2,4
INCFSZ	f, d	f增1，为0则跳过	1	0011	11df	ffff	无	2,4
IORWF	f, d	W和f作逻辑或运算	1	0001	00df	ffff	Z	2,4
MOVF	f, d	将f的内容送到目标寄存器	1	0010	00df	ffff	Z	2,4
MOVWF	f	将W的内容传送到f	1	0000	001f	ffff	无	1,4
NOP	—	空操作	1	0000	0000	0000	无	
RLF	f, d	对f执行带进位的循环左移	1	0011	01df	ffff	C	2,4
RRF	f, d	对f执行带进位的循环右移	1	0011	00df	ffff	C	2,4
SUBWF	f, d	f减去W	1	0000	10df	ffff	C、DC和Z	1,2,4
SWAPF	f, d	将f中的两个半字节进行交换	1	0011	10df	ffff	无	2,4
XORWF	f, d	W和f作逻辑异或运算	1	0001	10df	ffff	Z	2,4
面向位的文件寄存器操作								
BCF	f, b	将f中的某位清零	1	0100	bbbf	ffff	无	2,4
BSF	f, b	将f中的某位置1	1	0101	bbbf	ffff	无	2,4
BTFSC	f, b	检测f中的某位，为0则跳过	1	0110	bbbf	ffff	无	2
BTFSS	f, b	检测f中的某位，为1则跳过	1	0111	bbbf	ffff	无	2
立即数和控制操作								
ANDLW	k	立即数与W作逻辑与运算	1	1110	kkkk	kkkk	Z	
CALL	k	调用子程序	2	1001	kkkk	kkkk	无	1
CLRWDT	k	清零看门狗定时器	1	0000	0000	0100	$\overline{TO}$和$\overline{PD}$	
GOTO	k	无条件跳转	2	101k	kkkk	kkkk	无	
IORLW	k	立即数与W作逻辑或运算	1	1101	kkkk	kkkk	Z	
MOVLW	k	将立即数传送到W	1	1100	kkkk	kkkk	无	
OPTION	—	装载选项寄存器	1	0000	0000	0010	无	
RETLW	k	返回并将立即数传送到W	2	1000	kkkk	kkkk	无	
SLEEP	—	进入待机模式	1	0000	0000	0011	$\overline{TO}$和$\overline{PD}$	
TRIS	f	装载TRIS寄存器	1	0000	0000	0fff	无	3
XORLW	k	立即数与W作逻辑异或运算	1	1111	kkkk	kkkk	Z	

注 **1：** 除 GOTO 之外的任何写 PC 的指令都将把程序计数器的第 9 位强制为 0。请参见**第 4.7 节 “程序计数器”**。

2： 当 I/O 寄存器用自身内容修改自身时（例如：MOVF PORTB, 1），使用的值是出现在引脚上的值。例如，如果将一引脚配置为输入，虽然其对应数据锁存器中的值为 1，但此时若有外部器件将该引脚驱动为低电平，则被写回的数据值将是 0。

3： 指令 TRIS f（其中 f = 6）会将 W 寄存器的内容写入 PORTB 的三态锁存器。若写入的值为 1，将强制引脚为高阻态，并且禁止输出缓冲器。

4： 当对 TMR0 寄存器（并且 d = 1）执行这条指令时，如果将预分频器分配给 TMR0，则将其清零。

9.1 指令说明

ADDWF W 和 f 相加

语法：	[*标号*] ADDWF f, d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(W) + (f) → (目标寄存器)
受影响的状态位：	C、 DC 和 Z
说明：	将 W 寄存器的内容与 f 寄存器的内容相加。如果 d 等于 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

ANDLW 立即数与 W 作逻辑与运算

语法：	[*标号*] ANDLW k
操作数：	0 ≤ k ≤ 255
操作：	(W) .AND.(k) → (W)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与 8 位立即数 k 作逻辑与运算。结果存入 W 寄存器。

ANDWF W 和 f 作逻辑与运算

语法：	[*标号*] ANDWF f, d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(W) .AND.(f) → (目标寄存器)
受影响的状态位：	Z
说明：	W 寄存器与 f 寄存器作逻辑与运算。如果 d 等于 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

BCF 将 f 中的某位清零

语法：	[*标号*]BCF f, b
操作数：	0 ≤ f ≤ 31 0 ≤ b ≤ 7
操作：	0 → (f<b>)
受影响的状态位：	无
说明：	将寄存器 f 中的位 b 清零。

BSF 将 f 中的某位置 1

语法：	[*标号*] BSF f,b
操作数：	0 ≤ f ≤ 31 0 ≤ b ≤ 7
操作：	1 → (f<b>)
受影响的状态位：	无
说明：	将寄存器 f 的位 b 置 1。

BTFSC 检测 f 中的某位，为 0 则跳过

语法：	[*标号*] BTFSC f,b
操作数：	0 ≤ f ≤ 31 0 ≤ b ≤ 7
操作：	如果 (f<b>) = 0 则跳过
受影响的状态位：	无
说明：	如果 f 寄存器中的位 b 为 0，则跳过下一条指令。 如果 f 寄存器中的位 b 为 0，则丢弃当前指令执行时预取的下一条指令，转而执行一条 NOP 指令，从而使该指令成为双周期指令。

`

BTFSS	**检测 f 中的某位，为 1 跳过**
语法：	[*标号*] BTFSS f,b
操作数：	0 ≤ f ≤ 31 0 ≤ b < 7
操作：	如果 (f<b>)= 1 则跳过
受影响的状态位：	无
说明：	如果 f 寄存器中的位 b 为 1，则跳过下一条指令。 如果位 b 为 1，则丢弃在当前指令执行时所预取的下一条指令，转而执行一条 NOP 指令，从而使该指令成为双周期指令。

CLRW	**将 W 寄存器清零**
语法：	[*标号*] CLRW
操作数：	无
操作：	00h → (W) 1 → Z
受影响的状态位：	Z
说明：	W 寄存器被清零。全零标志位（Z）置 1。

`

CALL	**调用子程序**
语法：	[*标号*] CALL k
操作数：	0 ≤ k ≤ 255
操作：	(PC)+ 1 → 栈顶， k → PC<7:0>， (STATUS<6:5>) → PC<10:9> 0 → PC<8>
受影响的状态位：	无
说明：	调用子程序。首先，将返回地址（PC + 1）压入堆栈。8 位立即数地址被装入 PC<7:0>。将 STATUS<6:5> 装入 PC<10:9>。CALL 是双周期指令。

CLRWDT	**清零看门狗定时器**
语法：	[*标号*] CLRWDT
操作数：	无
操作：	00h → WDT 0 → WDT 预分频器 （如果分配） 1 → $\overline{TO}$ 1 → $\overline{PD}$
受影响的状态位：	$\overline{TO}$ 和 $\overline{PD}$
说明：	CLRWDT 指令复位 WDT。如果将预分频器分配给 WDT 而不是 Timer0，该指令还将复位该预分频器。 状态位 $\overline{TO}$ 和 $\overline{PD}$ 置 1。

CLRF	**将 f 清零**
语法：	[*标号*] CLRF f
操作数：	0 ≤ f ≤ 31
操作：	00h → (f) 1 → Z
受影响的状态位：	Z
说明：	寄存器 f 的内容被清零， Z 位置 1。

COMF	**f 取反**
语法：	[*标号*] COMF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	$(\bar{f})$ → (目标寄存器)
受影响的状态位：	Z
说明：	将寄存器 f 的内容取反。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

DECF **f 减 1**

语法：	[*标号*] DECF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(f) – 1 → (目标寄存器)
受影响的状态位：	Z
说明：	将寄存器 f 的内容减 1。如果 d 为 0，结果存储到 W 寄存器。如果 d 为 1，结果存回寄存器 f。

DECFSZ **f 减 1，为 0 则跳过**

语法：	[*标号*] DECFSZ f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(f) – 1 → (目标寄存器)； 结果为 0 时跳过
受影响的状态位：	无
说明：	将寄存器 f 的内容减 1。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。 如果结果为 0，则丢弃已经取指的指令而执行一条 NOP 指令，使该指令成为双周期指令。

GOTO **无条件跳转**

语法：	[*标号*] GOTO k
操作数：	0 ≤ k ≤ 511
操作：	k → PC<8:0> STATUS<6:5> → PC<10:9>
受影响的状态位：	无
说明：	GOTO 是无条件跳转指令。 9 位立即数地址被装入 PC<8:0>。 PC 高位从 STATUS<6:5> 装入。 GOTO 是双周期指令。

INCF **f 增 1**

语法：	[*标号*] INCF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(f) + 1 → (目标寄存器)
受影响的状态位：	Z
说明：	将寄存器 f 的内容增 1。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

INCFSZ **f 增 1，为 0 则跳过**

语法：	[*标号*] INCFSZ f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(f) + 1 → (目标寄存器)，结果为 0 时跳过
受影响的状态位：	无
说明：	将寄存器 f 的内容增 1。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。 如果结果为 0，则丢弃已经取指的指令而执行一条 NOP 指令，使该指令成为双周期指令。

IORLW **立即数与 W 作逻辑或运算**

语法：	[*标号*] IORLW k
操作数：	0 ≤ k ≤ 255
操作：	(W) .OR. k → (W)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与 8 位立即数 k 作逻辑或运算。结果存入 W 寄存器。

IORWF **将 W 和 f 作逻辑或运算**

语法： [*标号*] IORWF f,d

操作数： $0 \le f \le 31$
d ∈ [0,1]

操作： (W) .OR.(f) → (目标寄存器)

受影响的状态位： Z

说明： W 寄存器与 f 寄存器作逻辑或运算。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

MOVF **将 f 的内容送到目标寄存器**

语法： [*标号*] MOVF f,d

操作数： $0 \le f \le 31$
d ∈ [0,1]

操作： (f) → (目标寄存器)

受影响的状态位： Z

说明： 根据 d 的状态，将寄存器 f 的内容送入目标寄存器。如果 d 为 0，目标寄存器为 W 寄存器。如果 d 为 1，目标寄存器为寄存器 f。由于状态标志位 Z 受到指令结果的影响，d = 1 可用于检测文件寄存器。

MOVLW **将立即数传送到 W**

语法： [*标号*] MOVLW k

操作数： $0 \le k \le 255$

操作： k → (W)

受影响的状态位： 无

说明： 将 8 位立即数 k 装入 W 寄存器。“无关位”被汇编为 0。

MOVWF **将 W 的内容传送到 f**

语法： [*标号*] MOVWF f

操作数： $0 \le f \le 31$

操作： (W) → (f)

受影响的状态位： 无

说明： 将 W 寄存器中的数据传送到寄存器 f。

NOP **空操作**

语法： [*标号*] NOP

操作数： 无

操作： 空操作

受影响的状态位： 无

说明： 不执行任何操作。

OPTION **装载选项寄存器**

语法： [*标号*] OPTION

操作数： 无

操作： (W) → OPTION

受影响的状态位： 无

说明： 将 W 寄存的内容装入选项寄存器。

RETLW 返回并将立即数传送到 W

语法：	[*标号*] RETLW k
操作数：	0 ≤ k ≤ 255
操作：	k → (W) TOS → PC
受影响的状态位：	无
说明：	将 8 位立即数 k 装入 W 寄存器。栈顶内容 （返回地址）被装入程序计数器。这是一条双周期指令。

RLF 对 f 执行带进位的循环左移

语法：	[*标号*] RLF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	参见下面的说明
受影响的状态位：	C
说明：	将寄存器 f 的内容连同进位标志位一起左移 1 位。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

C ← 寄存器 'f' ←

RRF 对 f 执行带进位的循环右移

语法：	[*标号*] RRF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	参见下面的说明
受影响的状态位：	C
说明：	将寄存器 f 的内容连同进位标志位一起右移 1 位。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

→ C → 寄存器 f →

SLEEP 进入休眠模式

语法：	[*标号*] SLEEP
操作数：	无
操作：	00h → WDT， 0 → $\overline{\text{WDT}}$ 预分频器， 1 → $\overline{\text{TO}}$， 0 → $\overline{\text{PD}}$
受影响的状态位：	$\overline{\text{TO}}$、 $\overline{\text{PD}}$ 和 RBWUF
说明：	超时状态位 （$\overline{\text{TO}}$）位置 1。掉电状态位 （$\overline{\text{PD}}$）清零。 RBWUF 不受影响。 WDT 及其预分频器被清零。 振荡器停振，处理器进入休眠模式。欲知有关休眠的详细信息，请参见相关章节。

SUBWF f 减去 W

语法：	[*标号*] SUBWF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(f) – (W) → (目标寄存器)
受影响的状态位：	C、 DC 和 Z
说明：	从寄存器 f 中减去 W 寄存器的内容（采用 2 的补码方法进行运算）。如果 d 为 0，结果存储到 W 寄存器。如果 d 为 1，结果存回寄存器 f。

SWAPF 将 f 中的两个半字节交换

语法：	[*标号*] SWAPF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(f<3:0>) → (目标寄存器 <7:4>), (f<7:4>) → (目标寄存器 <3:0>)
受影响的状态位：	无
说明：	将寄存器 f 的高半字节和低半字节交换。如果 d 为 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

TRIS	装载 TRIS 寄存器
语法：	[*标号*] TRIS f
操作数：	f = 6
操作：	(W) → TRIS 寄存器 f
受影响的状态位：	无
说明：	使用 W 寄存器的内容装载 TRIS 寄存器 f （f = 6 或 7）。

XORLW	立即数与 W 作逻辑异或运算
语法：	[*标号*] XORLW k
操作数：	0 ≤ k ≤ 255
操作：	(W).XOR. k → (W)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与 8 位立即数 k 作逻辑异或运算。结果存入 W 寄存器。

XORWF	W 和 f 作逻辑异或运算
语法：	[*标号*] XORWF f,d
操作数：	0 ≤ f ≤ 31 d ∈ [0,1]
操作：	(W) .XOR.(f) → (目标寄存器)
受影响的状态位：	Z
说明：	将 W 寄存器的内容与 f 寄存器的内容作逻辑异或运算。如果 d 等于 0，结果存放在 W 寄存器中。如果 d 为 1，结果存回寄存器 f。

初稿

10.0 电气规范

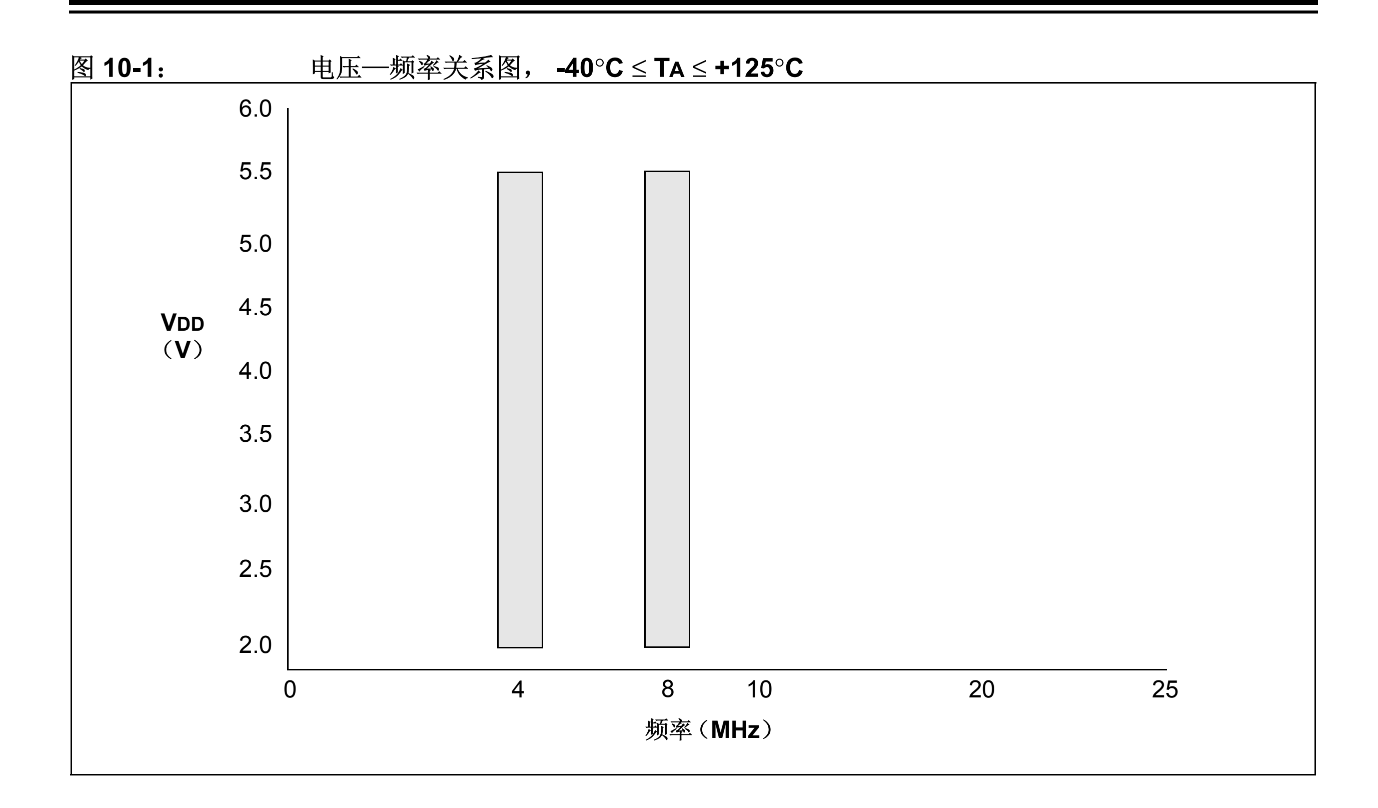
绝对最大值 (†)

环境温度 .... -40°C 至 +125°C
储存温度 .... -65°C 至 +150°C
V_{DD} 相对于 V_{SS} 的电压 .... 0 至 +6.5V
$\overline{MCLR}$ 相对于 V_{SS} 的电压 .... 0 至 +13.5V
所有其他引脚相对于 V_{SS} 的电压 .... -0.3V 至 (V_{DD}+ 0.3V)
总功耗 (1) .... 200 mW
V_{SS} 引脚的最大输出电流 .... 80 mA
V_{DD} 引脚的最大输出电流 .... 80 mA
输入钳位电流 I_{IK} （V_I < 0 或 V_I > V_{DD}） .... ±20 mA
输出钳位电流 I_{OK} （V_O < 0 或 V_O > V_{DD}） .... ±20 mA
任一 I/O 引脚的最大输出灌电流 .... 25 mA
任一 I/O 引脚的最大输出拉电流 .... 25 mA
I/O 端口的最大输出拉电流 .... 75 mA
I/O 端口的最大输出灌电流 .... 75 mA

注 1：功耗按如下公式计算：$P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$

† 注：如果器件工作条件超过“绝对最大值”，可能会对器件造成永久性损坏。上述值仅为运行条件极大值，我们建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下，其稳定性会受到影响。

图 10-1： 电压—频率关系图， -40°C ≤ TA ≤ +125°C



10.1 直流特性：PIC10F220/222 （工业级）

直流特性			标准工作条件 （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
D001	VDD	电源电压	2.0	—	5.5	V	请参见图 **10-1**
D002	VDR	**RAM** 数据保持电压 (2)	—	1.5*	—	V	器件处于休眠模式
D003	VPOR	**VDD** 起始电压确保能够产生上电复位信号	—	Vss	—	V	详情请参见第 **8.4** 节 “上电复位（**POR**）”
D004	SVDD	**VDD** 上升速率确保能够产生上电复位信号	0.05*	—	—	V/ms	详情请参见第 **8.4** 节 “上电复位（**POR**）”
D010	IDD	供电电流 (3)	— — — —	170 350 250 450	TBD TBD TBD TBD	μA μA μA μA	FOSC = 4 MHz， VDD = 2.0V FOSC = 4 MHz， VDD = 5.0V FOSC = 8 MHz， VDD = 2.0V FOSC = 8 MHz， VDD = 5.0V
D020	IPD	掉电电流 (4)	—	0.1	TBD	μA	VDD = 2.0V
D022	ΔIWDT	**WDT** 电流 (4)	—	1.0	TBD	μA	VDD = 2.0V
D024	ΔIADC	**A/D** 电流	—	80	TBD	μA	VDD = 2.0V

图注： TBD = 待定。

* 这些参数仅为特征值，未经测试。

注 **1：** “典型值”栏中的数据均为 25°C 条件下的值。该数据仅供设计参考而未经测试。

2： 该电压是休眠模式下保证不丢失 RAM 数据的最小 VDD。

3： 供电电流主要是由工作电压和频率一起决定的。其他因素，如总线负载、总线速率、内部代码执行模式和温度都对电流消耗有影响。

a) 在正常工作模式下，所有 IDD 测量的测试条件为：
所有 I/O 引脚均为三态，拉至 VSS； T0CKI = VDD，$\overline{MCLR}$ = VDD； WDT 的禁止和使能根据需要指定。

b) 在测量待机电流时，除了器件处于休眠模式之外，其他条件完全相同。

4： 掉电电流是在器件休眠时，所有 I/O 引脚处于高阻态并且连接到 VDD 或 VSS 时测得的。

10.2 直流特性：PIC10F220/222 （扩展级）

直流特性			标准工作条件 （除非另外声明） 工作温度 -40°C ≤ TA ≤ +125°C （扩展级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
D001	VDD	电源电压	2.0		5.5	V	请参见图 **10-1**
D002	VDR	**RAM 数据保持电压** (2)	—	1.5*	—	V	器件处于休眠模式
D003	VPOR	**VDD 起始电压**确保能够产生上电复位信号	—	Vss	—	V	详情请参见第 **8.4** 节 "上电复位（**POR**）"
D004	SVDD	**VDD 上升速率**确保能够产生上电复位信号	0.05*	—	—	V/ms	详情请参见第 **8.4** 节 "上电复位（**POR**）"
D010	IDD	供电电流 (3)	— — — —	170 350 250 450	TBD TBD TBD TBD	μA μA μA μA	FOSC = 4 MHz, VDD = 2.0V FOSC = 4 MHz, VDD = 5.0V FOSC = 8 MHz, VDD = 2.0V FOSC = 8 MHz, VDD = 5.0V
D020	IPD	掉电电流 (4)	—	0.1	TBD	μA	VDD = 2.0V
D022	ΔIWDT	**WDT 电流** (4)	—	1.0	TBD	μA	VDD = 2.0V
D024	ΔIADC	**A/D 电流**	—	80	TBD	μA	VDD = 2.0V

图注： TBD = 待定。

* 这些参数仅为特征值，未经测试。

注 **1：** "典型值"栏中的数据均为 25°C 条件下的值。该数据仅供设计参考而未经测试。

2： 该电压是休眠模式下保证不丢失 RAM 数据的最小 VDD。

3： 供电电流主要是由工作电压和频率一起决定的。其他因素，如总线负载、总线速率、内部代码执行模式和温度都对电流消耗有影响。

a) 在正常工作模式下，所有 IDD 测量的测试条件为：
所有 I/O 引脚均为三态，拉至 VSS； T0CKI = VDD，$\overline{MCLR}$ = VDD； WDT 的禁止和使能根据需要指定。

b) 在测量待机电流时，除了器件处于休眠模式之外，其他条件完全相同。

4： 掉电电流是在器件休眠时，所有 I/O 引脚处于高阻态并且连接到 VDD 或 VSS 时测得的。

表 10-1： 直流特性：PIC10F220/222 （工业级和扩展级）

直流特性			标准工作条件 （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级） 直流规范说明了工作电压 VDD 的范围				
参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
	VIL	输入低电压					
		I/O 端口：					
D030		带 TTL 缓冲器	Vss	—	0.8V	V	适用于 4.5 ≤ Vdd ≤ 5.5V
D030A			Vss	—	0.15 VDD	V	其他情况
D031		带施密特触发缓冲器	Vss	—	0.15 VDD	V	
D032		MCLR 和 T0CKI	Vss	—	0.15 VDD	V	
	VIH	输入高电压					
		I/O 端口：		—			
D040		带 TTL 缓冲器	2.0	—	VDD	V	4.5V ≤ Vdd ≤ 5.5V
D040A			0.25 VDD + 0.8	—	VDD	V	其他情况
D041		带施密特触发缓冲器	0.85 VDD	—	VDD	V	适用于整个 VDD 范围
D042		MCLR 和 T0CKI	0.85 VDD	—	VDD	V	
D070	IPUR	**GPIO 弱上拉电流 (3)**	TBD	250	TBD	μA	VDD = 5V 且 VPIN = VSS
	IIL	输入泄漏电流 (1,2)					
D060		I/O 端口	—	—	±1	μA	Vss ≤ VPIN ≤ VDD，引脚处于高阻态
D061		GP3/MCLR(4)	—	—	±30	μA	Vss ≤ VPIN ≤ VDD
D061A		GP3/MCLR(5)	—	—	±5	μA	Vss ≤ VPIN ≤ VDD
		输出低电压					
D080		I/O 端口	—	—	0.6	V	IOL = 8.5 mA 且 VDD = 4.5V, -40°C 至 +85°C
D080A			—	—	0.6	V	IOL = 7.0 mA 且 VDD = 4.5V, +85°C 至 +125°C
		输出高电压					
D090		I/O 端口 (2)	VDD - 0.7	—	—	V	IOH = -3.0 mA 且 VDD = 4.5V， -40°C 至 +85°C
D090A			VDD - 0.7	—	—	V	IOH = -2.5 mA 且 VDD = 4.5V， +85°C 至 +125°C
		输出引脚上的容性负载规范					
D101		所有 I/O 引脚	—	—	50*	pF	

图注： TBD = 待定。

† 除非另外声明，否则 “典型值” 栏中的数据均为 5V 和 25×C 下的值。这些参数仅供设计参考，未经测试。

* 这些参数仅供设计参考，未经测试。

注 1： MCLR/VPP 引脚上的泄漏电流主要由施加在该引脚上的电平决定。规定电平为正常工作条件下的电平。在不同的输入电压下可测得更高的泄漏电流。

2： 负电流定义为自引脚流出的电流。

3： 不包括 GP3。关于 GP3 请参见参数 D061 和 D061A。

4： 该规范适用于配置为外部 MCLR 的 GP3/MCLR 和配置为输入并使能了内部上拉电路的 GP3/MCLR。

5： 本规范在 GP3/MCLR 配置为输入并且禁用上拉电路的时候适用。 MCLR 电路的泄漏电流高于标准 I/O 逻辑。

表 **10-2**： 上拉电阻的范围

VDD （V）	温度 （°C）	最小值	典型值	最大值	单位
GP0/GP1					
2.0	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω
5.5	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω
GP3					
2.0	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω
5.5	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω

图注： TBD = 待定。
* 这些参数仅为特征值，未经测试。

10.3 时序参数符号和负载条件

可根据以下一种格式来创建时序参数符号：

1. TppS2ppS
2. TppS

T	
F 频率	T 时间

小写下标（pp）和它们的含义：

pp			
2	至	mc	MCLR
ck	CLKOUT	osc	振荡器
cy	周期时间	os	OSC1
drt	器件复位定时器	t0	T0CKI
io	I/O 端口	wdt	看门狗定时器

大写字母和它们的含义：

S			
F	下降	P	周期
H	高	R	上升
I	无效（高阻态）	V	有效
L	低	Z	高阻态

图 10-2： 负载条件

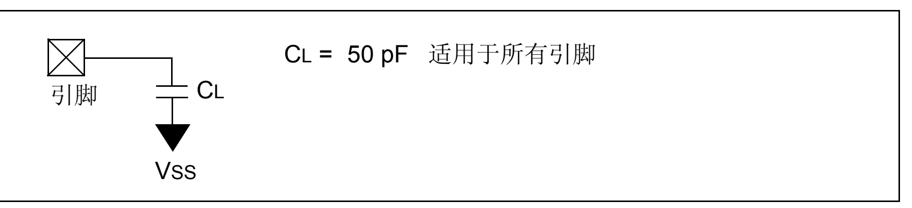


表 10-3： 校准的内部 RC 频率

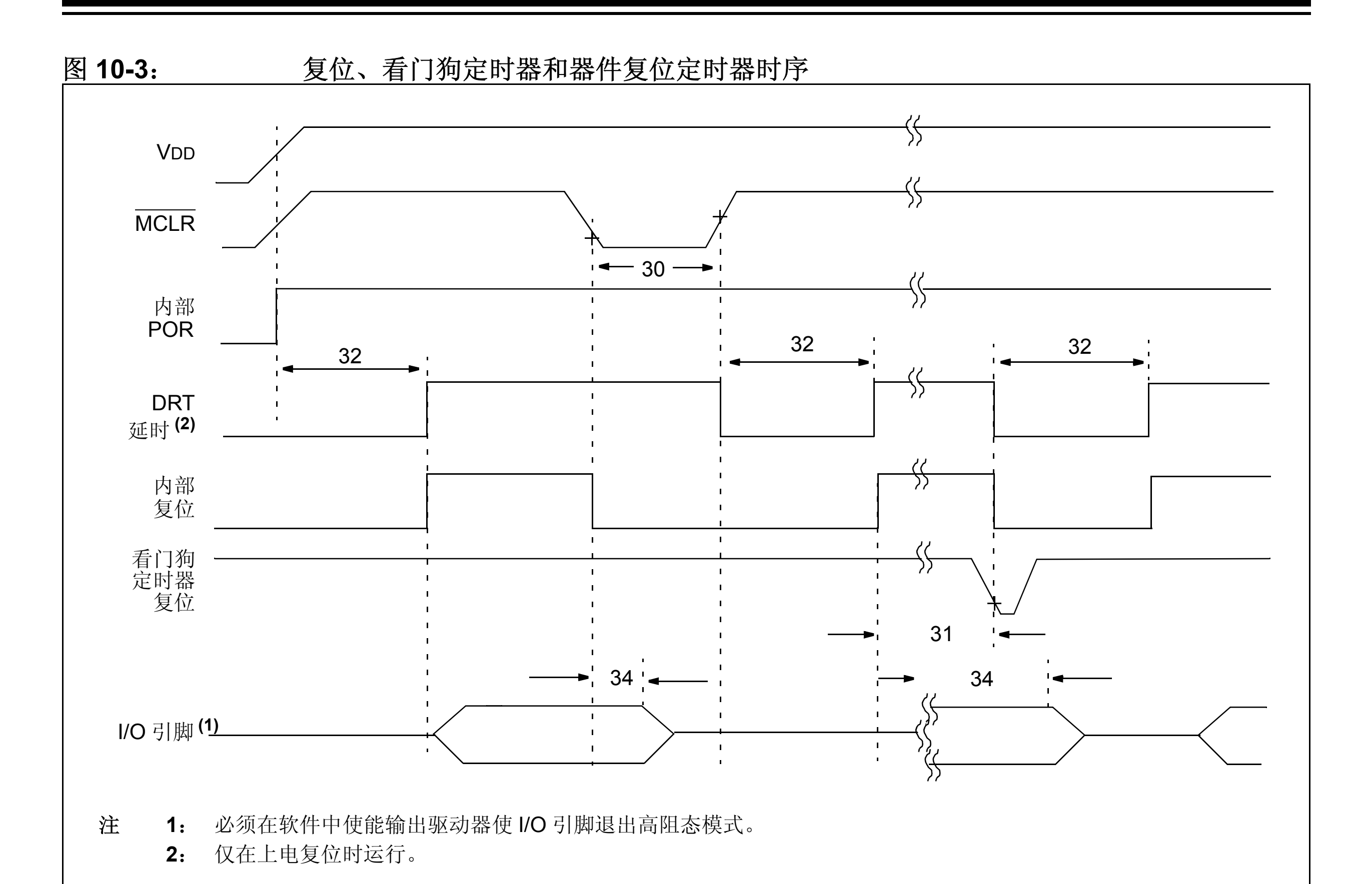
交流特性			标准工作条件（除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）					
参数编号	符号	特性	频率容差	最小值	典型值 †	最大值	单位	条件
F10	FOSC	经过校准的内部 INTOSC 频率 (1,2)	±1%	7.92	8	8.08	MHz	VDD 和温度待定
			±2%	7.84	8	8.16	MHz	2.5V ≤ VDD ≤ 5.5V 温度 0 - 85°C
			±5%	7.60	8	8.4	MHz	2.0V ≤ VDD ≤ 5.5V -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）

图注： TBD = 待定
* 这些参数仅为特征值，未经测试。
† 除非另外声明，否则“典型值”栏中的数据均在 5.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

注 1： 为了确保振荡器频率的容差，必须尽量靠近器件在 VDD 和 VSS 之间连接去耦电容。 建议并联 0.1 µF 和 0.01 µF 的电容。

2： 4 MHz 时钟由 8 MHz 振荡器产生。要获得 4 MHz 容差值，将相应的 8 MHz 时钟进行 2 分频。

图 10-3： 复位、看门狗定时器和器件复位定时器时序



注 1： 必须在软件中使能输出驱动器使 I/O 引脚退出高阻态模式。
2： 仅在上电复位时运行。

表 10-4： 复位、看门狗定时器和器件复位定时器

交流特性			标准工作条件 （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
30	TMCL	$\overline{\text{MCLR}}$ 脉冲宽度 （低电平）	2000*	—	—	ns	VDD = 5.0V
31	TWDT	看门狗定时器超时周期 （无预分频器）	9* 9*	18* 18*	30* 40*	ms ms	VDD = 5.0V （工业级） VDD = 5.0V （扩展级）
32	TDRT	器件复位定时器周期	0.5* 0.5*	1.125* 1.125*	2* 2.5*	ms ms	VDD = 5.0V （工业级） VDD = 5.0V （扩展级）
34	TIOZ	$\overline{\text{MCLR}}$ 低电平引起 I/O 高阻态	—	—	2000*	ns	

* 这些参数仅为特征值，未经测试。

注 1： 除非另外声明，否则 “典型值”栏中的数据均为 5V 和 25°C 下的值。这些参数仅供设计参考，未经测试。

 初稿

图 **10-4**： **TIMER0** 时钟时序

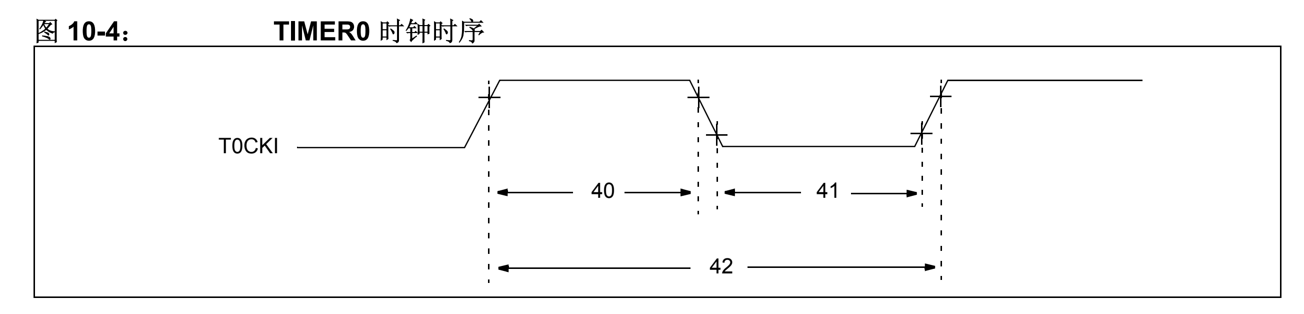


表 **10-5**： **TIMER0** 时钟要求

交流特性		标准工作条件 （除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C （工业级） -40°C ≤ TA ≤ +125°C （扩展级）						
参数编号	符号	特性		最小值	典型值 (1)	最大值	单位	条件
40	Tt0H	T0CKI 高电平脉冲宽度	无预分频器	0.5 TCY + 20*	—	—	ns	
			有预分频器	10*	—	—	ns	
41	Tt0L	T0CKI 低电平脉冲宽度	无预分频器	0.5 TCY + 20*	—	—	ns	
			有预分频器	10*	—	—	ns	
42	Tt0P	T0CKI 周期		20 或 TCY + 40* N	—	—	ns	较大者。 N = 预分频值 （1、 2、4、 ...，256）

* 这些参数仅为特征值，未经测试。

注 **1**： 除非另外声明，否则 “典型值”栏中的数据均为 5V 和 25°C 下的值。这些参数仅供设计参考，未经测试。

表 10-6： A/D 转换器特性

参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
A01	NR	分辨率	—	—	8	位	
A02	EABS	总绝对误差 *(1)	—	—	TBD	LSb	VDD = 5.0V
A03	EIL	积分误差	—	—	TBD	LSb	VDD = 5.0V
A04	EDL	微分误差	—	—	TBD	LSb	不丢失编码至 8 位 VDD = 5.0V
A05	EFS	满量程范围	2.2*	—	5.5*	V	VDD
A06	EOFF	失调误差	—	—	TBD	LSb	VREF = 5.0V
A07	EGN	增益误差	—	—	TBD	LSb	VREF = 5.0V
A10	—	单调性	—	保证 (2)	—	—	VSS ≤ VAIN ≤ VREF+
A25	VAIN	模拟输入电压	—	VDD	—	V	
A30	ZAIN	模拟电压源阻抗推荐值	—	—	10	kΩ	

* 这些参数仅为特征值，未经测试。

† 除非另外声明，否则 “典型值”栏中的数据均在 5.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

注 1： 总绝对误差包括积分误差、微分误差、失调误差和增益误差。

2： A/D 转换结果不会因输入电压的增加而减小，并且不会丢失编码。

3： VREF 电流来自作为参考电压输入的外部 VREF 或 VDD 引脚。

4： 当处于关闭状态时， A/D 转换器只消耗泄漏电流。掉电电流包括任何来自于 A/D 模块的泄漏电流。

 初稿

11.0 开发支持

一系列硬件及软件开发工具对 PICmicro® 单片机提供支持：

- 集成开发环境
  - MPLAB® IDE 软件
- 汇编器 / 编译器 / 链接器
  - MPASM™ 汇编器
  - MPLAB C18 和 MPLAB C30 C 编译器
  - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
  - MPLAB ASM30 汇编器 / 链接器 / 库
- 模拟器
  - MPLAB SIM 软件模拟器
- 仿真器
  - MPLAB ICE 2000 在线仿真器
  - MPLAB ICE 4000 在线仿真器
- 在线调试器
  - MPLAB ICD 2
- 器件编程器
  - PICSTART® Plus 开发编程器
  - MPLAB PM3 器件编程器
  - PICkit™ 2 开发编程器
- 低成本演示和开发板及评估工具包

11.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16 位单片机市场提供了前所未有的易于使用的软件开发平台。MPLAB IDE 是基于 Windows® 操作系统的应用软件，包括：

- 一个包含所有调试工具的图形界面
  - 模拟器
  - 编程器 （单独销售）
  - 仿真器 （单独销售）
  - 在线调试器 （单独销售）
- 具有彩色上下文代码显示的全功能编辑器
- 多项目管理器
- 内容可直接编辑的可定制式数据窗口
- 高级源代码调试
- 可视化器件初始化程序，便于进行寄存器的初始化
- 鼠标停留在变量上进行查看的功能
- 通过拖放把变量从源代码窗口拉到观察窗口
- 丰富的在线帮助
- 集成了可选的第三方工具，如 HI-TECH 软件 C 编译器和 IAR C 编译器

MPLAB IDE 可以让您：

- 编辑源文件 （汇编语言或 C 语言）
- 点击一次即可完成汇编 （或编译）并将代码下载到 PICmicro MCU 仿真器和模拟器工具中 （自动更新所有项目信息）
- 可使用如下各项进行调试：
  - 源文件 （汇编语言或 C 语言）
  - 混合汇编语言和 C 语言
  - 机器码

MPLAB IDE 在单个开发范例中支持使用多种调试工具，包括从成本效益高的模拟器到低成本的在线调试器，再到全功能的仿真器。这样缩短了用户升级到更加灵活而功能更强大的工具时的学习时间。

11.2 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于所有的 PICmicro MCU。

MPASM 汇编器可生成用于 MPLINK 目标链接器的可重定位目标文件、Intel® 标准 HEX 文件、详细描述存储器使用状况和符号参考的 MAP 文件、包含源代码行及生成机器码的绝对 LST 文件以及用于调试的 COFF 文件。

MPASM 汇编器具有如下特征：

- 集成在 MPLAB IDE 项目中
- 用户定义的宏可简化汇编代码
- 对多用途源文件进行条件汇编
- 允许完全控制汇编过程的指令

11.3 MPLAB C18 和 MPLAB C30 C 编译器

MPLAB C18 和 MPLAB C30 代码开发系统是完全的 ANSI C 编译器，分别适用于 Microchip 的 PIC18 系列单片机及 dsPIC30F、dsPIC33 和 PIC24 系列数据信号控制器。这些编译器可提供其他编译器并不具备的强大的集成功能和出众的代码优化能力，且使用方便。

为便于源代码调试，编译器提供了针对 MPLAB IDE 调试器的优化符号信息。

11.4 MPLINK 目标链接器 / MPLIB 目标库管理器

MPLINK 目标链接器包含了由 MPASM 汇编器、MPLAB C18 C 编译器产生的可重定位目标。通过使用链接器脚本中的指令，它还可链接预编译库中的可重定位目标。

MPLIB目标库管理器管理预编译代码库文件的创建和修改。当从源文件调用库中的一段子程序时，只有包含此子程序的模块被链接到应用中。这样可使大型库在许多不同应用中被高效地利用。

目标链接器 / 库管理器具有如下特征：

- 高效地连接单个的库而不是许多小文件
- 通过将相关的模块组合在一起来增强代码的可维护性
- 只要列出、替换、删除和抽取模块，便可灵活地创建库

11.5 MPLAB ASM30 汇编器、链接器和库管理器

MPLAB ASM30 汇编器为 dsPIC30F 器件提供转换自符号汇编语言的可重定位机器码。 MPLAB C30 C 编译器使用该汇编器生成目标文件。汇编器产生可重定位目标文件之后，可将这些目标文件存档，或与其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特征：

- 支持整个 dsPIC30F 指令集
- 支持定点数据和浮点数据
- 命令行界面
- 丰富的指令集
- 灵活的宏语言
- MPLAB IDE 兼容性

11.6 MPLAB SIM 软件模拟器

MPLAB SIM 软件模拟器在指令级对 PICmicro MCU 和 dsPIC® DSC 进行模拟，使得用户可以在 PC 主机的环境下进行代码开发。对于任何给定的指令，用户均可对数据区进行检查或修改，并通过各种触发机制来产生激励。可以将各寄存器的情况记录在文件中，以便进行进一步地运行时分析。跟踪缓冲器和逻辑分析器的显示使模拟器还能记录和跟踪程序的执行、 I/O 的动作、大部分的外设及内部寄存器的状况。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C18 和 MPLAB C30 C 编译器以及MPASM和MAPLAB ASM30 汇编器的符号调试。该软件模拟器可用于在硬件实验室环境外灵活地开发和调试代码，是一款完美且经济的软件开发工具。

 初稿

11.7 MPLAB ICE 2000 高性能在线仿真器

MPLAB ICE 2000 在线仿真器旨在为产品开发工程师提供一整套用于 PICmicro 单片机的设计工具。 MPLAB ICE 2000 在线仿真器的软件控制由 MPLAB 集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 2000 是全功能仿真器系统，它具有增强的跟踪、触发和数据监控功能。处理器模块可插拔，使系统可轻松进行重新配置以适应各种不同处理器的仿真需要。MPLAB ICE 2000 在线仿真器的架构允许对其进行扩展以支持新的 PICmicro 单片机。

MPLAB ICE 2000 在线仿真器系统设计为一款实时仿真系统，该仿真系统具备通常只有昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft® Windows® 32 位操作系统可使这些功能在一个简单而统一的应用中得到很好的利用。

11.8 MPLAB ICE 4000 高性能在线仿真器

MPLAB ICE 4000 在线仿真器旨在为产品开发工程师提供一整套用于高端 PICmicro MCU 和 dsPIC DSC 的设计工具。 MPLAB ICE 4000 在线仿真器的软件控制由 MPLAB 集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 4000 是高级的仿真系统，除具备 MPLAB ICE 2000 的所有功能外，它还增加了适用于 dsPIC30F 和 PIC18XXXX 器件的仿真存储容量以及高速性能。该仿真器的先进特性包括复杂触发和定时功能及高达 2 Mb 的仿真存储容量。

MPLAB ICE 4000 在线仿真系统设计为一款实时仿真系统，该仿真系统具备通常只有在更加昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft Windows 32 位操作系统可使这些功能在一个简单而统一的应用程序中得以很好的利用。

11.9 MPLAB ICD 2 在线调试器

Microchip 的在线调试器 MPLAB ICD 2 是一款功能强大而成本低廉的运行时开发工具，通过 RS-232 或高速 USB 接口与 PC 主机相连。该工具基于闪存 PICmicro MCU，可用于开发本系列及其他 PICmicro MCU 和 dsPIC DSC。 MPLAB ICD 2 使用了闪存器件中内建的在线调试功能。该功能结合 Microchip 的在线串行编程（In-Circuit Serial Programming™， ICSP™）协议，可在 MPLAB 集成开发环境的图形用户界面上提供成本效益很高的在线闪存调试。这使设计人员可通过设置断点、单步运行以及对变量、CPU 状态以及外设寄存器进行监视的方法实现源代码的开发和调试。其全速运行特性可对硬件和应用进行实时测试。MPLAB ICD 2 还可用作某些 PICmicro 器件的开发编程器。

11.10 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款通用的、符合 CE 规范的器件编程器，其可编程电压设置在 VDDMIN 和 VDDMAX 之间时可靠性最高。它有一个用来显示菜单和错误信息的大 LCD 显示器（128 x 64），以及一个支持各种封装类型的可拆卸模块化插槽装置。编程器标准配置中带有一根 ICSP™ 电缆。在单机模式下， MPLAB PM3 器件编程器不必与 PC 相连即可对 PICmicro 器件进行读取、验证和编程。在该模式下它还可设置代码保护。MPLAB PM3 通过 RS-232 或 USB 电缆连接到 PC 主机上。MPLAB PM3 具备高速通信能力以及优化算法，可对存储器很大的器件进行快速编程，它还采用 SD/MMC 卡用作文件存储及数据安全应用。

11.11 PICSTART Plus 开发编程器

PICSTART Plus开发编程器是一款易于使用而成本低廉的原型编程器。它通过 COM （RS-232）端口与 PC 相连。 MPLAB 集成开发环境软件使得该编程器的使用简便、高效。PICSTART Plus 开发编程器支持采用 DIP 封装的大部分 PICmicro 器件，其引脚数最多可达 40 个。引脚数更多的器件，如 PIC16C92X 和 PIC17C76X，可通过连接一个转接插槽来获得支持。PICSTART Plus 开发编程器符合 CE 规范。

11.12 PICkit 2 开发编程器

PICkit™ 2 开发编程器是一个低成本编程器，通过其易于使用的接口可对众多 Microchip 的低档、中档和 PIC18F 系列闪存单片机进行编程。PICkit 2 入门工具包中包含一个有实验布线区的开发板、十二堂系列课程、软件和 HI-TECH 的 PICC Lite C 编译器，有助于用户快速掌握 PIC® 单片机的使用。这一工具包为使用 Microchip 功能强大的中档闪存系列单片机进行编程、评估和应用开发，提供了所需的一切。

11.13 演示、开发和评估板

有许多演示、开发和评估板可用于各种 PICmicro MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于测试和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、 RS-232 接口、 LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外， Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、 KEELOQ® 数据安全产品 IC、CAN、 IrDA®、 PowerSmart® 电池管理、 SEEVAL® 评估系统、 Σ–Δ ADC、流速传感器，等等。

有关演示、开发和评估工具包的完整列表，请查阅 Microchip 公司网页 （www.microchip.com）以及最新的 *"Product Selector Guide（产品选型指南）"* （DS00148）。

12.0 直流和交流特性图表

当前没有图表。

注：

13.0 封装信息

13.1 封装标识信息

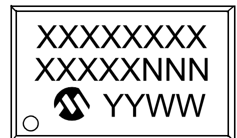
6 引脚 SOT-23

XXNN

示例

CH17

8 引脚 PDIP



示例

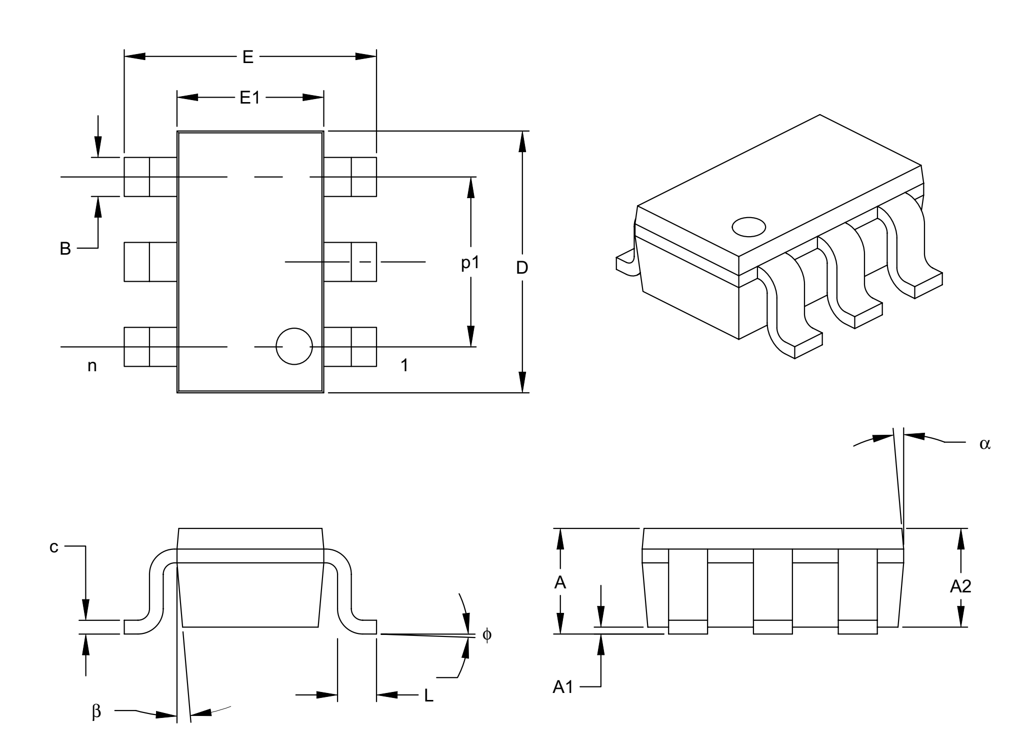
10F222/P
017
0510

图注：	XX...X	客户信息
	Y	年份代码 （日历年的最后一位数字）
	YY	年份代码 （日历年的最后两位数字）
	WW	星期代码 （一月一日的星期代码为 “01”）
	NNN	以字母数字排序的追踪代码
	(e3)	雾锡 （Matte Tin， Sn）的 JEDEC 无铅标志
	*	表示无铅封装。JEDEC 无铅标志（(e3)）标示于此种封装的外包装上。

注： Microchip 元器件编号如果无法在同一行内完整标注，将换行标出，因此会限制表示客户信息的字符数。

* 标准 PICmicro 器件标识由 Microchip 元器件编号、年份代码、星期代码和追踪代码组成。若 PICmicro 器件标识超出上述内容，需支付一定的附加费用。请向当地的 Microchip 销售办事处了解确认。对于 QTP 器件，任何特殊标记的费用都已包含在 QTP 价格中。

6 引脚塑封小型晶体管 （CH 或 OT）（SOT-23）



单位		英寸*			毫米		
尺寸范围		最小	正常	最大	最小	正常	最大
引脚数	n		6			6	
引脚间距	p		.038			0.95	
外部引脚间距	p1		.075			1.90	
顶端到固定面高度	A	.035	.046	.057	0.90	1.18	1.45
塑模封装厚度	A2	.035	.043	.051	0.90	1.10	1.30
基座到固定面高度	A1	.000	.003	.006	0.00	0.08	0.15
总宽度	E	.102	.110	.118	2.60	2.80	3.00
塑模封装宽度	E1	.059	.064	.069	1.50	1.63	1.75
总长度	D	.110	.116	.122	2.80	2.95	3.10
底足长度	L	.014	.018	.022	0.35	0.45	0.55
底足倾角	φ	0	5	10	0	5	10
底足厚度	c	.004	.006	.008	0.09	0.15	0.20
底足宽度	B	.014	.017	.020	0.35	0.43	0.50
塑模顶部锥度	α	0	5	10	0	5	10
塑模底部锥度	β	0	5	10	0	5	10

*控制参数

注：

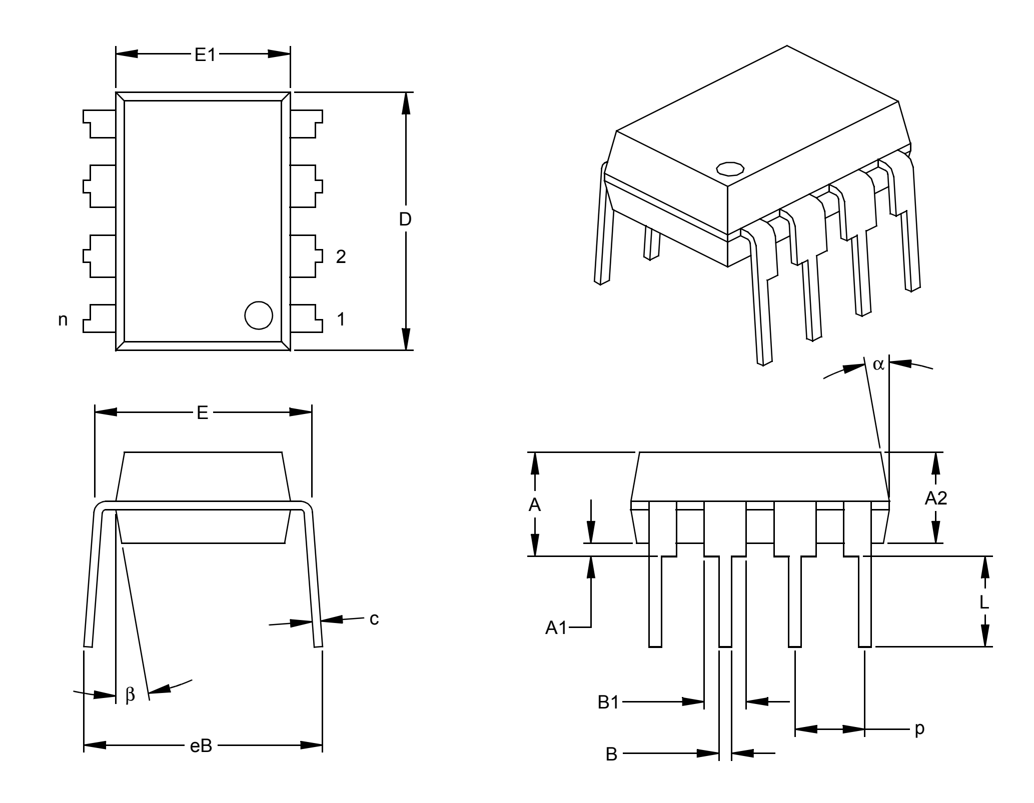
尺寸D和E1不包括塑模的毛边或突起。塑模每侧的毛边或突起不得超过0.005英寸（0.127毫米）。

等同于JEITA（以前称为EIAJ）号：SC-74A

图号：C04-120

初稿

8 引脚塑封双列直插式封装 （P）——主体 300 mil （PDIP）



单位		英寸 *			毫米		
尺寸范围		最小	正常	最大	最小	正常	最大
引脚数	n		8			8	
引脚间距	p		.100			2.54	
顶端到固定面高度	A	.140	.155	.170	3.56	3.94	4.32
塑模封装厚度	A2	.115	.130	.145	2.92	3.30	3.68
塑模底面到固定面高度	A1	.015			0.38		
肩到肩宽度	E	.300	.313	.325	7.62	7.94	8.26
塑模封装宽度	E1	.240	.250	.260	6.10	6.35	6.60
总长度	D	.360	.373	.385	9.14	9.46	9.78
引脚尖到固定面高度	L	.125	.130	.135	3.18	3.30	3.43
引脚厚度	c	.008	.012	.015	0.20	0.29	0.38
引脚上部宽度	B1	.045	.058	.070	1.14	1.46	1.78
引脚下部宽度	B	.014	.018	.022	0.36	0.46	0.56
总排列间距 §	eB	.310	.370	.430	7.87	9.40	10.92
塑模顶部锥度	α	5	10	15	5	10	15
塑模底部锥度	β	5	10	15	5	10	15

* 控制参数
§ 重要特性

注：
尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸（0.254 毫米）。

等同于 JEDEC 号：MS-001
图号：C04-018

注：

索引

 初稿

MICROCHIP 网站

Microchip 网站（www.microchip.com）为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息：

- **产品支持**——数据手册和勘误表、应用笔记和样本程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题（FAQ）、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip 的变更通知客户服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请登录 Microchip 网站 www.microchip.com，点击“变更通知客户（Customer Change Notification）”服务后按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师（FAE）
- 技术支持

客户应联系其代理商、代表或应用工程师（FAE）寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 http：//support.microchip.com 获得网上技术支持。

读者反馈表

我们努力为您提供最佳文档，以确保您能够成功使用 Microchip 产品。如果您对文档的组织、条理性、主题及其他有助于提高文档质量的方面有任何意见或建议，请填写本反馈表并传真给我公司 TRC 经理，传真号码为 86-21-5407-5066。

请填写以下信息，并从下面各方面提出您对本文档的意见。

致: TRC 经理　　　　总页数 ________

关于: 读者反馈

发自: 姓名 ______________________________

公司 ______________________________

地址 ______________________________

国家 / 省份 / 城市 / 邮编 ______________________________

电话: (______) ____________________　　传真: (______) ____________________

应用（选填）:

您希望收到回复吗？是____ 否____

器件: PIC10F220/222　　文献编号: DS41270A_CN

问题

1. 本文档中哪些部分最有特色？

____

____

2. 本文档是否满足了您的软硬件开发要求？如何满足的？

____

____

3. 您认为本文档的组织结构便于理解吗？如果不便于理解，那么问题何在？

____

____

4. 您认为本文档应该添加哪些内容以改善其结构和主题？

____

____

5. 您认为本文档中可以删减哪些内容，而又不会影响整体使用效果？

____

____

6. 本文档中是否存在错误或误导信息？如果存在，请指出是什么信息及其具体页数。

____

____

7. 您认为本文档还有哪些方面有待改进？

____

____

产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或销售办事处联系。

部件编号	X	/XX	XXX
器件	温度范围	封装	模式

器件	PIC10F220(1) 和 PIC10F222(1)； VDD 范围：2.0V 至 5.5V
温度范围	I = -40°C 至 +85°C （工业级） E = -40°C 至 +125°C （扩展级）
封装	OT = SOT 和 6 引脚 （无铅） P = 300 mil PDIP 和 8 引脚 （无铅）
模式	特殊要求

示例：

a) PIC10F220 - I/P = 工业级温度，PDIP 封装（无铅）

b) PIC10F222 - T-I/OT = 工业级温度，SOT 封装（无铅）

注 1： SOT 封装只有卷带式。



全球销售及服务网点

美洲

公司总部 Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277
技术支持：
http://support.microchip.com
网址：www.microchip.com

亚特兰大 Atlanta
Alpharetta, GA
Tel: 1-770-640-0034
Fax: 1-770-640-0307

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo
Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣何塞 San Jose
Mountain View, CA
Tel: 1-650-215-1444
Fax: 1-650-961-0286

加拿大多伦多 Toronto
Mississauga, Ontario, Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

中国 – 北京
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 – 成都
Tel: 86-28-8676-6200
Fax: 86-28-8676-6599

中国 – 福州
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

中国 – 香港特别行政区
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 – 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 – 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 – 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 – 深圳
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 – 顺德
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

中国 – 武汉
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 – 西安
Tel: 86-29-8833-7250
Fax: 86-29-8833-7256

台湾地区 – 高雄
Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 – 台北
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 – 新竹
Tel: 886-3-572-9526
Fax: 886-3-572-6459

亚太地区

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

印度 India - New Delhi
Tel: 91-11-5160-8631
Fax: 91-11-5160-8632

印度 India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama
Tel: 81-45-471-6166
Fax: 81-45-471-6122

韩国 Korea - Gumi
Tel: 82-54-473-4301
Fax: 82-54-473-4302

韩国 Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Penang
Tel: 60-4-646-8870
Fax: 60-4-646-5086

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

丹麦 Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

02/16/06